



ระบบการจดจำเสียงพูด  
SPEECH RECOGNITION SYSTEM



ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

สาขาวิศวกรรมการวัดคุมอุตสาหกรรม

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษานั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ปีการศึกษา 2537

034753

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปริญญาโทปีการศึกษา 2537


ภาควิชา เทคโนโลยีการวิศวกรรม

คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง ระบบการจดจำเสียงพูด

ผู้จัดทำ

- |               |              |           |
|---------------|--------------|-----------|
| 1. นายวิรัตน์ | ศิริชวงค์    | 34 106335 |
| 2. นางสาวหทัย | เลี่ยมเพชร   | 41 09430  |
| 3. นายอักรพล  | อัสวปรมิตชัย | 34 109497 |

  
..... อาจารย์ที่ปรึกษา  
(อาจารย์เกษตร์ ศิริสันติสัมฤทธิ์)

## ระบบการจดจำเสียงพูด

นายวิรัตน์ กิริตวิวงศ์

นางสาวหทัย เลี่ยมเพชร

นายอัครพล อัครปรมิตชัย

อาจารย์เกษตร์ สิริสันติสัมฤทธิ์ อาจารย์ที่ปรึกษา

### บทคัดย่อ

ปัจจุบัน เทคโนโลยีทางการติดต่อสื่อสารระหว่างมนุษย์กับเครื่องจักรถูกพัฒนาไปอย่างคืบเนื่องเทคนิคของการใช้สัญญาณเสียงพูด ซึ่งได้รับความสนใจอย่างกว้างขวางนอกเหนือไปจากวิธีการอื่น ๆ เนื่องจากเป็นวิธีการที่ง่ายต่อการใช้งาน ในปริยญาณิพนธ์ฉบับนี้จะเป็นโครงการระบบการจดจำเสียงพูด ซึ่งจะกล่าวถึงส่วนประกอบต่างๆ ของเสียง วิธีการสุ่มสัญญาณ การแปลงฟูเรียร์อย่างรวดเร็ว ระบบการจดจำเสียงพูดโดยใช้วิธีการหาค่าเฉลี่ยทางสถิติแบบวิธีการหาค่าเบี่ยงเบนมาตรฐาน ระบบการวิเคราะห์เสียง ซึ่งนำเอาข้อมูลเสียงที่บันทึกมาเปรียบเทียบกับ ข้อมูลเสียงที่เข้ามา เพื่อหาค่าเปอร์เซ็นต์ของความเป็นไปได้ โดยในโครงการนี้นำมาใช้กับเสียงพูด 4 คำต่างๆ กัน ซึ่งผลการทดลองที่ได้อยู่ในขั้นที่น่าพอใจ

# SPEECH RECOGNITION SYSTEM

WIRAT KEERATICHOOWONG

HATHAI LIAMPET

AKARAPON ASAVAPORAMITTHAI

KASET SIRISUNTISUMRIT ADVISOR

## ABSTRACT

Nowaday technology of the communication between human and machine have been continuous developed. Technique of using speech signal has been worldwidelly interested because of the easy using. In this thesis would talk about voice component, sampling method, fast fourier transform, speech recognition system which we use the statistical average by using standard deviation method, Speech analysis system for comparison between storage data and input data to calculate the possibility. This project use 4 isolated words and the results are satisfied.

## สารบัญ

บทนำ	1
บทที่ 1 ลักษณะของเสียงพูด	2
บทที่ 2 ทฤษฎีการสุ่ม	4
บทที่ 3 การแปลงฟาสต์ฟูเรียร์	13
บทที่ 4 การจดจำเสียงพูด	22
บทที่ 5 แนวทางการออกแบบฮาร์ดแวร์	45
บทที่ 6 ทฤษฎีที่เกี่ยวข้องกับงานวิจัย	60
บทที่ 7 การทดลองและผลการทดลอง	71
บทที่ 8 การใช้โปรแกรม	84
ภาคผนวก 1	94
ภาคผนวก 2	132
หนังสืออ้างอิง	134

## บทนำ

ในปัจจุบัน มนุษย์เราพยายามหาแนวทางในการติดต่อสื่อสารระหว่างมนุษย์กับเครื่องจักร เพื่อให้ได้วิธีการที่ดีที่สุด ซึ่งอาจเป็นวิธีการที่เร็วที่สุดในการติดต่อ หรือเป็นวิธีการที่สามารถนำข้อมูลคำสั่งไปประมวลผลได้เร็วที่สุด หรืออาจเป็นวิธีการที่สะดวกต่อมนุษย์ได้มากที่สุด ทั้งนี้แล้วแต่ความต้องการของการนำไปใช้กับชนิดของงานการผลิตและบริการเพื่อตอบสนองความต้องการของมนุษย์ให้มีความพึงพอใจสูงสุด เสียงพูดของมนุษย์จึงเป็นอีกแนวทางหนึ่งที่ถูกพิจารณานำมาทดลองเพื่อตอบสนองตามความต้องการของมนุษย์เนื่องจากเป็นวิธีการที่สะดวกต่อมนุษย์ที่จะใช้ระบบการติดต่อ จากการที่ได้ศึกษาถึงโครงการทดลองอื่นๆที่ผ่านมา พบว่า ลักษณะของโครงงานมักใช้วิธีการในรูปแบบของการวิเคราะห์ในรูปแบบของโดเมนเวลา (time domain) โดยเก็บข้อมูลเป็นขนาดของข้อมูลเฉลี่ยที่เข้ามาว่ามีขนาดใหญ่แค่ไหน แต่เนื่องจากเสียงพูดซึ่งเป็นสัญญาณอนาล็อก ประกอบด้วยคุณสมบัติหลายๆตัวมาประกอบกัน เช่น ความถี่หลายๆความถี่หลายๆเฟส หลายๆขนาด เป็นต้น ดังนั้นการวิเคราะห์แต่เพียงขนาดอย่างเดียวจะไม่สามารถให้ค่าได้ถูกต้องแม่นยำ โครงงานนี้จึงศึกษาวิธีอื่นๆ นำมาใช้ร่วมกับการวิเคราะห์ในรูปแบบเดิม เพื่อเพิ่มประสิทธิภาพในการวิเคราะห์ให้ดียิ่งขึ้น

## 1.2 การเกิดของเสียง (Speech Production)

ขั้นตอนการเกิดของเสียงแบ่งได้เป็น 3 ขั้นตอนใหญ่ ๆ คือ

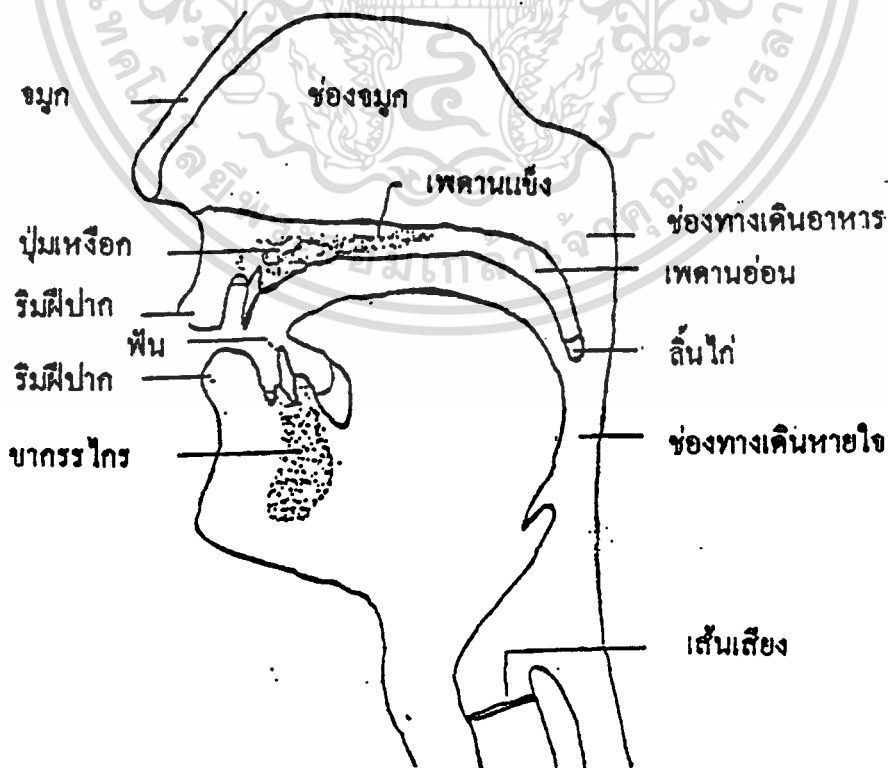
1. ขั้นเริ่มต้น (Initiation) ขั้นตอนนี้เป็นขั้นตอนที่ลมเริ่มถูกขับออกปอดเพื่อเข้าสู่ขั้นตอนที่สองต่อไป

2. ขั้นตอนคัดแปลงลมที่เส้นเสียง (Phonation) เป็นขั้นตอนที่ลมจากปอดจะผ่านมายังหลอดลมและกล่องเสียง ซึ่งที่กล่องเสียงนี้ เส้นเสียงจะทำหน้าที่เป็น ลิ้นเปิด ปิด ทำให้เกิดเสียงได้สองชนิด คือ เสียงก้อง (Voiced Sounds) และ เสียงที่ไม่ก้อง (Unvoiced Sounds) อวัยวะที่ใช้ในตอนนี้คือ ส่วนที่ต่อจากปอดขึ้นมาจนถึงริมฝีปาก

เสียงที่เกิดจากขั้นตอนดังกล่าว จะถูกแบ่งเป็น 2 ประเภท ใหญ่ ๆ คือ

1. เสียงก้อง (Voiced Sounds) เกิดเนื่องจากการออกเสียงในลักษณะที่เส้นเสียงถูกดึงเข้ามาใกล้กันจนเกือบปิดช่องทางลมเสียดสีกัน ลมที่ดันขึ้นมาจากปอดจะทำให้เส้นเสียงสั่น ลมที่ออกมาไม่สะดวก เพราะต้องบีบตัวผ่านช่องแคบเป็นจังหวะจึงทำให้เกิดเป็นเสียงขึ้น เรียกว่า "เสียงก้อง"

2. เสียงไม่ก้อง (Unvoiced Sounds) เป็นการออกเสียงในขณะที่เส้นเสียงยังเปิดกว้าง โดยเปิดช่องระหว่างเส้นเสียงหรือช่องคอหอย (Glottis) ให้ลมหายใจผ่านเข้าออกสะดวก เรียกเสียงเหล่านี้ว่า "เสียงไม่ก้อง"



เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ไว้เพื่อใช้ในการออกเสียงพูดให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## บทที่ 2 ทฤษฎีการสุ่ม (Sampling Theory)

ในความเป็นจริงนั้น ระบบที่ต่อเนื่อง input/output และ ตัวแปรอื่นๆ ที่เกี่ยวข้องในระบบทั้งหมด จะมีค่าที่เปลี่ยนแปลงไปตามเวลาแบบต่อเนื่อง ซึ่งเราสามารถที่จะหาค่าต่างๆ โดยการวัดได้ ณ เวลาใดๆ แต่ในระบบแบบไม่ต่อเนื่องที่มีช่วงเวลาการทำงานเป็นช่วงๆ ที่แน่นอนนั้น เราอาจสามารถหาค่าต่างๆ ในช่วงเวลาใดเวลาหนึ่งได้เป็นเพียงค่าที่ใกล้เคียงกับค่าจริงๆ ในระบบต่อเนื่องเท่านั้น เพราะถ้าจะให้มันมีค่าเท่ากับในระบบต่อเนื่องได้ก็ต่อเมื่อมีการทำงานในลักษณะที่มีช่วงเวลาการทำงานที่สั้นมากจนใกล้เคียงค่าศูนย์ ซึ่งหมายถึงข้อมูลที่ใช้ต้องมีค่าเป็นอนันต์ด้วย เพราะฉะนั้นถ้าเราจะนำเอาเครื่องคอมพิวเตอร์ ซึ่งมีหลักการทำงานแบบไม่ต่อเนื่องมาใช้กับงานแบบระบบต่อเนื่องจึงเป็นการยากลำบากและไม่เหมาะสม ดังนั้นจึงมีความจำเป็นที่สัญญาณของระบบแบบต่อเนื่องนั้น จำต้องถูกทำการสุ่ม (sampling) ตามช่วงเวลาแบบไม่ต่อเนื่องนั้นๆ โดยการสุ่มสัญญาณข้อมูล และทำการเปลี่ยนแปลงสัญญาณทางอนาลอก ให้เป็นข้อมูลเชิงตัวเลขหรือดิจิทัลอนั้น เราเรียกว่าการดิจิไตซ์ จากที่กล่าวถึงทั้งหมดนี้ นั่น เรา รู้จักกันในรูปแบบของการแปลงสัญญาณอนาลอกให้เป็นสัญญาณดิจิทัล ส่วนในทางตรงข้ามกับที่กล่าวถึงมานี้คือการแปลงสัญญาณดิจิทัลให้เป็นอนาลอก หรือการอนาไลซ์นั่นเอง

สัญญาณที่ถูกนำเข้ามาผ่านระบบทำการสุ่มบางทีก็มีความหมายเป็นลำดับ (sequence) แล้วแต่ว่าจะเป็นลำดับเข้าหรือลำดับออก โดยธรรมชาติแล้วสัญญาณใดๆ นั้นจะมีความยาวเป็นอนันต์ ดังนั้นเราจึงใช้เวกเตอร์เป็นคำใช้แยกข้อมูลให้มีความยาวที่แน่นอน ซึ่งจะทำการวิเคราะห์นั้น สามารถทำได้ง่ายขึ้นโดยการคำนวณด้วยระบบดิจิทัล

จากทฤษฎีดังกล่าว ทำให้เราสามารถอย่างได้ว่าข้อมูลเชิงตัวเลขที่เราสุ่มมาเพื่อเป็นตัวแทนสัญญาณทางอนาลอกต่าง ๆ นั้นย่อมมีจุดที่สุ่มที่แน่นอน หรือเราสามารถใช้อัตราสุ่มข้อมูลเชิงตัวเลขต่าง ๆ ทางอนาลอกได้เสมอและมีจำนวนที่แน่นอน เราเรียกความแตกต่างระหว่างข้อมูลทางตัวเลขที่เราสามารถกำหนดได้นี้ว่าข้อมูลทางควันไทซ์

การสุ่มข้อมูลแบบนี้อาจจะเกิดความคลาดเคลื่อนอยู่บ้าง เราเรียกว่าความผิดพลาดทางควันไทซ์ (quantizing error) ซึ่งความผิดพลาดทางควัน ไตซ์นี้เป็นปัญหาในเรื่องของสัญญาณรบกวน ซึ่งเป็นเรื่องปกติสำหรับการทำงานในระบบดิจิทัล และมักก่อให้เกิดความไม่เที่ยงตรงต่อระบบที่ทำงานเกี่ยวกับสัญญาณดิจิทัล

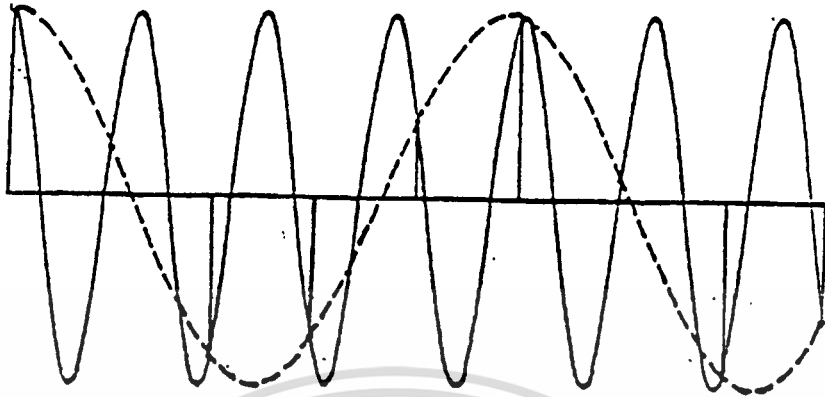
ในระบบที่เรากำลังศึกษานี้ จะใช้ช่วงเวลาในการสุ่มข้อมูลแต่ละครั้งเท่ากันเสมอ ดังนั้นอัตราการสุ่มข้อมูลของระบบจะมีค่าเท่ากับส่วนกลับของเวลาที่แตกต่างกันระหว่างการสุ่มข้อมูลแต่ละครั้ง

ในการสุ่มข้อมูลจากสัญญาณทางอนาลอกใดๆ ควรจะประมาณอัตราการสุ่มไว้ให้สูงพอสมควร เพราะถ้าใช้อัตราการสุ่มที่ค่อนข้างต่ำจนเกินไป อาจจะก่อให้เกิดปัญหาในระหว่างที่ทำการสุ่มข้อมูลจากสัญญาณต่อเนื่องขึ้นได้ เพราะในสัญญาณนั้นๆอาจมีข้อมูลที่สำคัญมาก และอาจก่อให้เกิดการเหลื่อมล้ำของข้อมูลที่ทำการสุ่มขึ้นมาได้ ผลที่ได้คือเมื่อทำการแปลงข้อมูลที่สุ่มมาได้กลับไปเป็นรูปของสัญญาณทางอนาลอก แล้วจะได้สัญญาณที่มีความผิดเพี้ยนจากสัญญาณต้นฉบับเดิมมาก แต่ถ้าเราใช้อัตราการสุ่มที่สูงเกินไป ซึ่งอาจจะมากเกินไปจนเกินไป จะทำให้เกิดความสิ้นเปลืองเนื้อที่ของหน่วยความจำโดยใช่เหตุ จะเห็นได้ว่าการสุ่มที่มากเกินไปก็ไม่ใช่วิธีที่ดีนัก

ในทางทฤษฎีการสุ่มแล้วนั้น การสุ่มข้อมูลแบบต่อเนื่องกำหนดให้มีการใช้อัตราการสุ่มอย่างน้อยที่สุดที่ยอมรับได้นี้ว่า อัตราไนควิสต์ (NYQUIST RATE) โดยถ้าเราต้องการรายละเอียดของสัญญาณสูง ก็จะต้องเลือกอัตราการสุ่มที่มากกว่าอัตราไนควิสต์ หรือมากกว่า 2 เท่าขึ้นไป ของความถี่สัญญาณสูงสุดที่นำมาทำการสุ่ม เช่น ถ้าสัญญาณที่จะนำมาทำการสุ่มมีความถี่ของสัญญาณเท่ากับ 3-4 กิโลเฮิร์ต อัตราการสุ่มที่เหมาะสมที่สุดจะเท่ากับ 8-9 กิโลเฮิร์ต

ในระบบที่มีการสุ่มน้อยกว่า NYQUIST RATE นั้น เราเรียกว่า Undersampling สำหรับสัญญาณที่ถูกสุ่มแบบ Undersampling นั้น จะได้ผลลัพธ์ที่มีความถี่เท่ากับ ผลต่างของความถี่ของอัตราการสุ่ม กับความถี่ของสัญญาณที่ป้อนเข้ามา ซึ่งจะมีความผิดเพี้ยนมากแสดงดังรูปที่ 2.1

ส่วนในระบบที่อัตราการสุ่มที่มากเกินไป (Oversampling) นั้นอาจมากเกินไปจนเกินไป ทำให้มีความสิ้นเปลืองเนื้อที่ของหน่วยความจำมาก และยังลำบากต่อการข้อมูลที่ได้ออกไปทำการวิเคราะห์เปรียบเทียบกับอีกด้วย

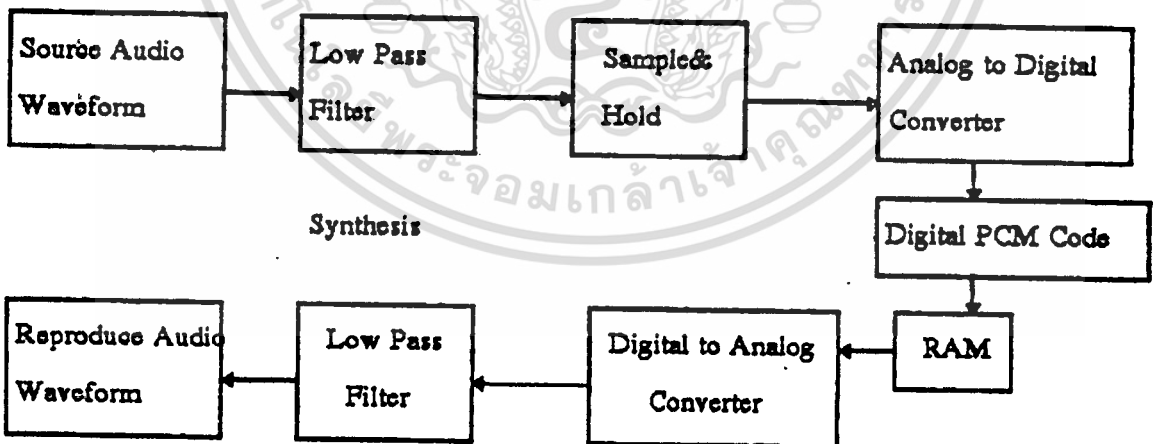


รูปที่ 2.1 แสดงตัวอย่างของการกำหนดอัตราการสุ่มต่ำกว่ามาตรฐาน

เส้นหนา แสดงถึง สัญญาณอินพุตที่มีความถี่ประมาณ 795 เฮิรตซ์

เส้นตรงทึบ แสดงถึง การสุ่มที่มีความถี่ประมาณ 1 กิโลเฮิรตซ์

เส้นประ แสดงถึง สัญญาณเอาต์พุตที่ได้ออกมา มีความถี่เท่ากับ ผลต่างของ อัตราสุ่ม กับสัญญาณอินพุต มีค่าประมาณ  $1000 - 795 = 205$  เฮิรตซ์



รูปที่ 2.2 แสดงไคอะแกรมของระบบการวิเคราะห์และการสังเคราะห์เสียงโดยวิธี PCM

คามหลักการแล้ว วงจรกรองความถี่จะต้องมีค่ามากกว่าครึ่งหนึ่งของอัตราสุ่มที่เรากำหนดไม่ควรให้จุดที่ทำการ out off นั้นทำการ out off เร็วเกินไป ควรเลือกให้การตอบสนองของวงจรกรองความถี่ผ่านนั้นลดลงอย่างช้าๆ จะดีกว่ามาก

วิธีการแปลงสัญญาณอนาล็อกให้เป็นสัญญาณดิจิทัลนั้นมีอยู่หลายวิธีการ แต่โดยหลักวิธีแล้วมีหลักการในรูปแบบการทำงานที่ค่อนข้างคล้ายกัน โดยหลักการพื้นฐานที่เราได้นำเอามาใช้ในที่นี้ และยังเป็นที่ยอมรับคือ แบบ Pulse Code Modulation (PCM)

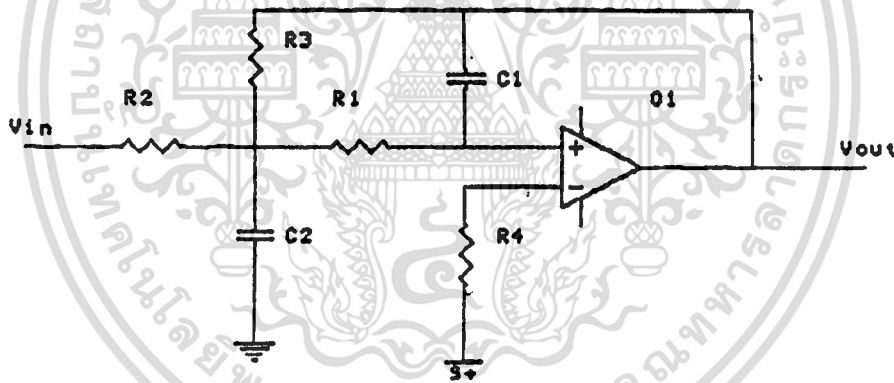
### 2.1 การเข้ารหัสแบบพัลส์โคดมอดูเลชัน (Pulse Code Modulation : PCM)

PCM คือ วิธีการที่ใช้โดยทั่วไปในการแทนสัญญาณอนาล็อกให้อยู่ในรูปแบบสัญญาณแบบดิจิทัล สำหรับจุดใดๆบนสัญญาณอนาล็อกที่ต่อเนื่องกันจะถูกแทนได้เป็นข้อมูลทางดิจิทัลค่าหนึ่งในระบบเลขฐานสอง เมื่อใดที่ค่าหนึ่งบนสัญญาณอนาล็อกถูกทำการวัดค่าหนึ่งๆจะถูกนำมาเปรียบเทียบว่าจะต้องแทนด้วยเลขใดคอะไร ในเลขฐานสอง ซึ่งเรียกว่าพัลส์โค้ด ถ้าหากว่าค่านั้นอยู่ในช่วงระหว่างสองค่าในเลขฐานสอง จะถือว่าค่าเลขฐานสองที่ใกล้เคียงกันมากที่สุดจะเป็นโค้ดสำหรับแทนสัญญาณอนาล็อกที่เข้าไป วิธีการเหล่านี้เรียกว่า ควันไทซ์เซชัน (Quantization) การแบ่งคลื่นออกเป็นช่วงย่อยๆ โดยแต่ละช่วงจะถูกกำหนดให้เท่ากับเลขฐานสองค่าหนึ่ง ดังนั้น ในการวิเคราะห์สัญญาณเสียงใดๆเมื่อสัญญาณเสียงผ่านกระบวนการเหล่านี้ข้อมูลที่เป็นพัลส์โค้ดจะถูกส่งออกมาอย่างต่อเนื่อง ผลที่ได้คือสัญญาณพัลส์โค้ดมอดูเลชัน แต่เพราะสัญญาณเสียงที่จะนำมาวิเคราะห์และทำการบันทึกไว้นั้นจะถูกนำมาเก็บในรูปแบบของสัญญาณดิจิทัล ดังนั้น การที่จะเอาสัญญาณเสียงนั้นๆมาผ่านกระบวนการต่างๆ เราจึงต้องทำการปรับแต่งสัญญาณนั้นๆให้เหมาะสมเสียก่อน ซึ่งได้ทำการแสดงวิธีการต่างๆดังในรูปที่ 2.2 ซึ่งได้แสดงถึงแผนผังการทำงานของเครื่องสังเคราะห์สัญญาณเสียง หรือ Speech Synthesizer ซึ่งเครื่องนี้จะทำการแปลงสัญญาณเสียงที่เราต้องการที่จะเก็บบันทึกไว้ให้อยู่ในรูปแบบของข้อมูลทางดิจิทัลที่เราเรียกว่า พัลส์โค้ดมอดูเลชัน (Pulse code modulation : PCM)

จากรูปที่ 2.2 สามารถอธิบายการทำงานได้ว่า เมื่อมีสัญญาณเสียงเข้ามา บางส่วนของสัญญาณอาจมีความถี่สูงเกินไป ซึ่งทำให้ค่าผลต่างระหว่างระดับสัญญาณสองจุดใดๆมีค่ามากทำให้เกิดการผิดพลาดขึ้นได้ จึงต้องมีการกรองความถี่สูงออกโดยการผ่านวงจรกรองความถี่ต่ำผ่าน (Low Pass Filter)

## 2.2 วงจรกรองความถี่ต่ำผ่าน (Low Pass Filter : LPF)

จากรูปที่ 2.2 จะเห็นตัวอุปกรณ์ประเภทวงจรกรองความถี่ต่ำ ซึ่งเป็นวงจรที่ใช้ในการกรองความถี่ข้อมูลมีระดับสูงจนเกินไปเกินกว่าค่าที่เราจะสนใจ หรือเกินกว่าระดับของการใช้งานโดยปกติ ทำให้ผลต่างระหว่างระดับสัญญาณ 2 จุดใดๆ ในขณะที่ทำการสุ่มมีค่ามากเกินกว่าค่าสูงสุดของพัลส์โค้ดที่เราได้กำหนดไว้ มักทำให้เกิดความผิดเพี้ยนของสัญญาณ (compliance error) จึงต้องผ่านวงจร Low Pass Filter เพื่อลดปัญหา ยกตัวอย่าง เช่น ระดับความถี่ที่เสียงคนเราปกติมักมีค่าอยู่ในช่วง 3-4 กิโลเฮิร์ต แต่ในบางครั้งความถี่ที่เราพูดอาจจะมีความถี่ถึง 5 กิโลเฮิร์ต เกิดขึ้นได้บ้าง ดังนั้นเมื่อความถี่ข้อมูลถูกส่งเข้ามาผ่านตัววงจรกรองความถี่ต่ำผ่าน ส่วนของข้อมูลที่มีความถี่เกินกว่า 4 กิโลเฮิร์ต ก็จะถูกทำการ detect เอาไว้ได้ ส่วนข้อมูลที่มีความถี่ต่ำกว่าก็จะสามารถผ่านไปได้ ในรูปที่ 2.3 จะเป็นการแสดงให้เห็นถึงลักษณะของวงจรกรองความถี่ต่ำผ่าน



รูปที่ 2.3 วงจรกรองความถี่ต่ำผ่าน

## 2.3 ความผิดเพี้ยนของสัญญาณ (Compliance Error)

ขณะที่ทำการสุ่มสัญญาณที่มีค่าความถี่มากเกินกว่าค่าสูงสุดของพัลส์โค้ดที่เราได้กำหนดไว้ ซึ่งมักทำให้เกิดความผิดเพี้ยนของสัญญาณที่เราเรียกว่า Compliance Error โดยค่าความถี่มีค่ามากเกินกว่าครึ่งหนึ่งของความถี่ของการสุ่มสัญญาณ อาจทำให้เกิดการผิดเพี้ยนอันเนื่องมาจากแถบคลื่นซ้อนกัน (Aliasing Distortion) ขณะที่มีการทำให้สัญญาณเดิมนั้นกลับคืนมา แต่อย่างไรก็ตามการผ่านตัว Low Pass Filter ทางด้านรับนั้น ในทางปฏิบัติไม่สามารถที่จะกำจัดความถี่ที่สูงกว่าค่าสูงสุดของพัลส์โค้ดได้ และฮาร์โมนิคต่างๆดังกล่าวเหล่านี้จะสอดแทรกเข้าเอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นุญาตให้นำไปเผยแพร่บนฐานการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



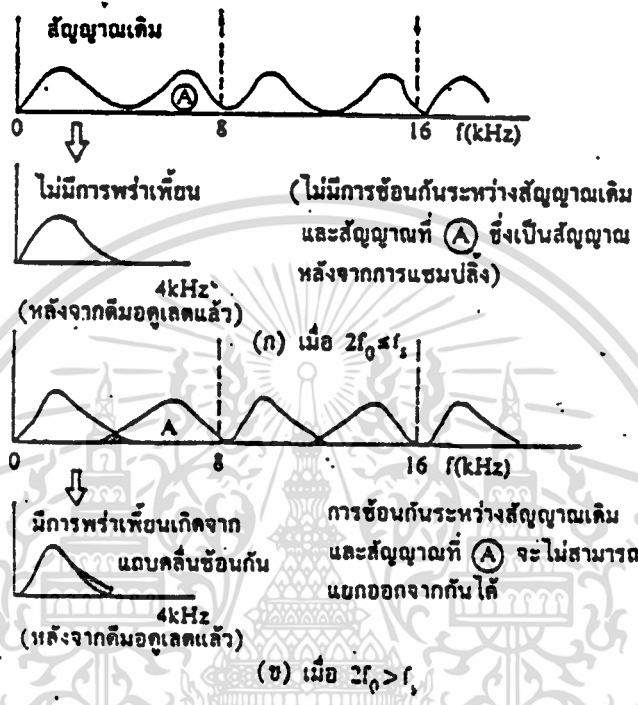
ไปปะปนอยู่กับสัญญาณที่ได้จากการแปลงกลับ หรือ คีมอดูเลท (Demodulate) จึงเป็นผลให้เกิดการผิดเพี้ยนที่เรียกว่าการผิดเพี้ยนเนื่องจากการสอดแทรก (Interpolating Distortion) ความผิดเพี้ยนทั้งสองแบบแสดงได้ดังรูปที่ 2.4 และ 2.5 ตามลำดับ

ดังนั้น เราจึงต้องนำสัญญาณที่เราต้องการจะวิเคราะห์และบันทึกมาผ่านวงจรกรองความถี่ต่ำ ซึ่งจะสามารถลดปัญหาเหล่านี้ได้ จากสัญญาณที่ได้จะถูกมาเข้าวงจรส่วนที่เรียกว่า วงจรสุ่มและหน่วงสัญญาณ (Sample & Hold circuit) เหตุที่ต้องใช้วงจรนี้เพราะ ในวงจรนี้เราได้เลือกใช้อุปกรณ์ที่ทำการแปลงสัญญาณอนาล็อกให้เป็นสัญญาณทางดิจิทัลที่ทำงานแบบ Successive approximation ซึ่งอุปกรณ์แปลงสัญญาณนี้ จะทำการแปลงสัญญาณออกมาทีละบิตแล้วกลับไปเช็คสัญญาณอินพุทใหม่อีก แล้วทำการแปลงไล่ไปจนครบทุกบิตจึงถือว่าเสร็จการทำงานไป 1 ครั้ง ถ้าหากไม่มีวงจรสุ่มและหน่วงสัญญาณไว้จะทำให้สัญญาณทางอินพุทของ ADC (Analog to Digital Converter) ไม่ใช่สัญญาณเดิม ในระหว่างกระบวนการทำการเปลี่ยนข้อมูลเหล่านี้กลับไปเป็นสัญญาณเสียงมักมีความผิดเพี้ยนอยู่เสมอ จากข้อมูลทางดิจิทัล ที่ได้รับออกมาแต่ละไบต์จะนำมาผ่านกระบวนการทางซอฟต์แวร์โดยไมโครโปรเซสเซอร์ เพื่อทำการแปลงสัญญาณออกมาในรูปของพัลส์โค้ด แล้วนำข้อมูลมาเก็บไว้ในหน่วยความจำ มาถึงจุดนี้คือเสร็จสิ้นขั้นตอนในการวิเคราะห์สัญญาณเสียงเพื่อเก็บบันทึกในรูปของสัญญาณดิจิทัลแบบพัลส์โค้ดมอดูเลชัน (Pulse Code Modulation)

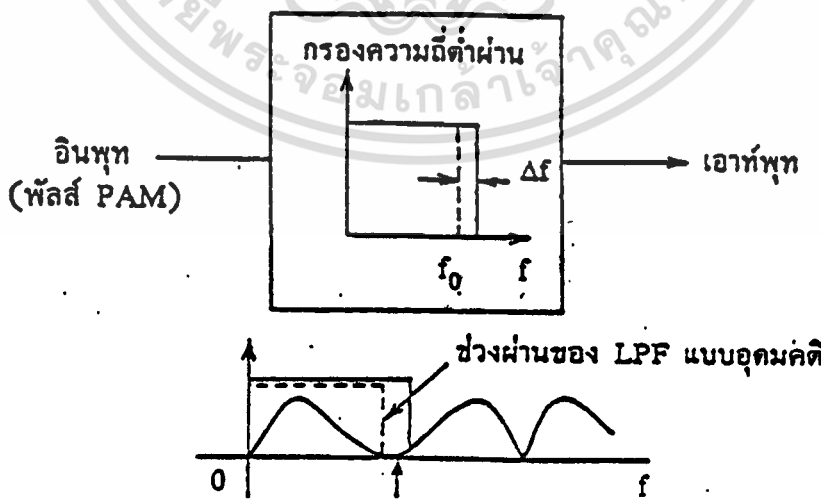
สำหรับขั้นตอนในการสังเคราะห์สัญญาณเสียงก็คือ การนำเอาข้อมูลพัลส์โค้ดที่มีอยู่ในหน่วยความจำมาทำการแปลงกลับให้เป็นสัญญาณเสียงโดยนำมาผ่านกระบวนการแปลงสัญญาณเหล่านั้นกลับออกมาอยู่ในรูปของข้อมูล 8 บิตโดยวิธีการทางซอฟต์แวร์ จากนั้นจึงนำเอาข้อมูลนั้นๆ มาผ่านวงจรแปลงสัญญาณดิจิทัลให้เป็นสัญญาณอนาล็อก แต่สัญญาณที่ออกมาจากวงจรแปลงสัญญาณดังกล่าวมักจะไม่เรียบหรือไม่มีความเป็นเชิงเส้นในลักษณะสัญญาณทางอนาล็อกเท่าที่ควร จึงมีความจำเป็นที่จะต้องนำเอาวงจรกรองความถี่ต่ำมาใช้ เพื่อให้สัญญาณมีความเป็นเชิงเส้นอย่างสมบูรณ์ที่สุด จึงเสร็จทุกขั้นตอนของการทำงานของระบบการวิเคราะห์และสังเคราะห์เสียงโดยวิธีพัลส์โค้ดมอดูเลชัน

หลังจากการสุ่มข้อมูลขนาดของแอมพลิจูดที่ได้จะเรียกว่า ควอนไทซ์ (Quantize) ซึ่งมีขนาดแบบไม่ต่อเนื่อง สัญญาณใหม่ที่ได้ี้จะเป็นสัญญาณแบบคร่าวๆของสัญญาณเดิม ซึ่งสัญญาณทั้งสองย่อมไม่เหมือนกันทุกส่วน ทำให้เกิดความผิดพลาดขึ้น ความแตกต่างระหว่างสองสัญญาณนี้เราเรียกว่า Quantizing error ในระบบการส่งข้อมูลแบบพัลส์โค้ดมอดูเลชัน สัญญาณรบกวนซึ่งเกิดจากควอนไทซ์ (Quantizing Noise) เป็นสัญญาณรบกวนที่สำคัญที่สุด ต้องระวังอย่างมาก

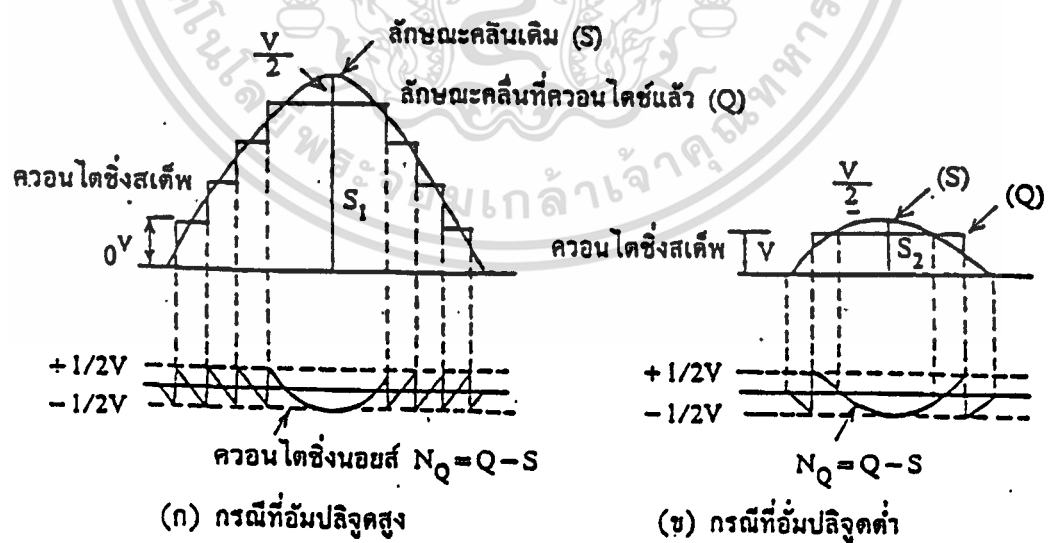
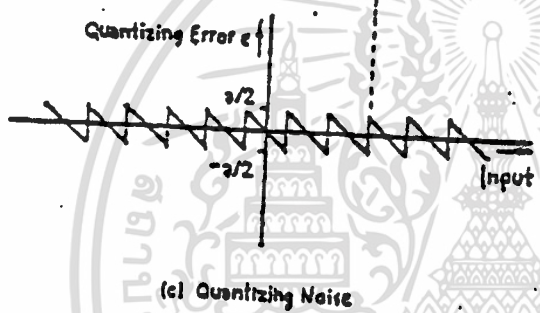
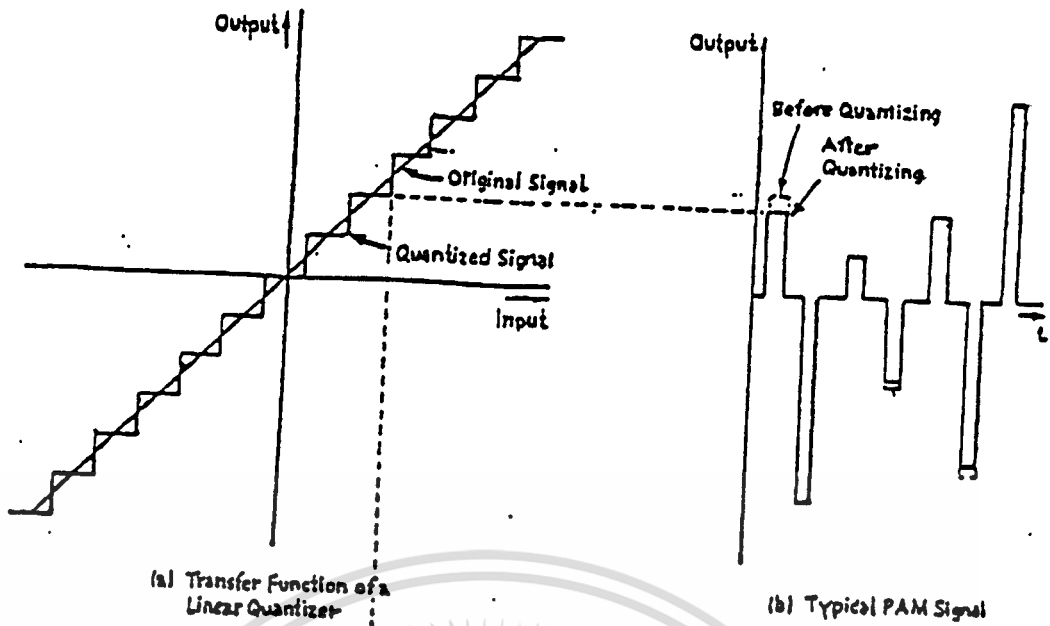
สัญญาณรบกวนที่เกิดจากการควอนไทซ์นั้น สามารถลดความผิดพลาดได้โดยการลดขนาดของช่วงควอนไทซ์ลง หรือเพิ่มระดับให้มากขึ้นไปอีก โดยเป็นการปรับเพิ่มหรือลดขนาดขั้นของการควอนไทซ์ให้มากขึ้นหรือลดลง (เปรียบเทียบกับกรปรับขึ้นบันไดโดยให้มีขนาดขั้นเดียวสำหรับเด็ก และมีขนาดขั้นสูงกว่าสำหรับผู้ใหญ่) ซึ่งจะกล่าวในหัวข้อต่อไป



รูปที่ 2.4 แสดงการผิดเพี้ยนที่เกิดจากแถบคลื่นซ้อนกัน



รูปที่ 2.5 แสดงการผิดเพี้ยนที่เกิดจากการสอดแทรก

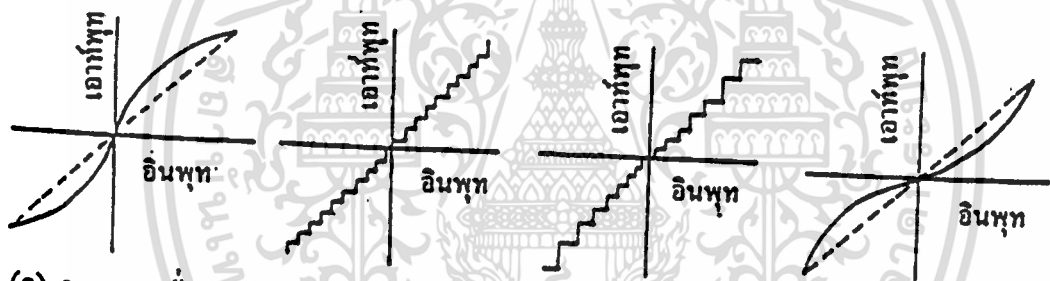


$$\frac{V/2}{S_1} < \frac{V}{S_2}$$

รูปที่ 2.6 แสดงการควอนไทซ์สัญญาณอนาล็อก และความผิดเพี้ยนที่เกิดจากการควอนไทซ์ เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น เมื่ออนุญาตให้เผยแพร่โดยไม่เสียค่าใช้จ่าย การนำเอกสารนี้ไปใช้โดยไม่ได้รับอนุญาตถือว่าผิดกฎหมาย และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

#### 2.4 การแก้ไขจากความผิดพลาดทางควอนไทซ์อันเนื่องจากสัญญาณรบกวน

สำหรับวงจรแบ่งระดับแบบไม่เท่ากัน (Non-linear Quantizing) นั้น มี มาตรฐานหลักๆ อยู่ 2 แบบ คือ แบบ  $\mu$ -Law และแบบ A-Law โดยทั้ง 2 ลักษณะที่กล่าวถึงนี้มีลักษณะการแบ่งแบบคอมแพนดิง (Companding) ซึ่งการแบ่งแบบนี้จะสามารถลดควอนไทซ์ซิงนอยส์ (Quantizing Noise) ที่ไม่สามารถหลีกเลี่ยงได้ โดยการลดควอนไทซ์ซิงอินเทอร์วัล (Quantizing Interval) หรือเป็นการเพิ่มจำนวนระดับนั่นเอง แต่เมื่อเพิ่มจำนวนระดับขึ้นแล้วจำนวนบิตที่ใช้ก็จะมากขึ้นด้วย จึงต้องใช้ความเร็วในการส่งสัญญาณดิจิทัลที่สูงขึ้น Companding นั้นเป็นชื่อรวมของวงจรคอมเพรสเซอร์ (Compressor) ซึ่งใช้สำหรับด้านส่ง กับ วงจรเอ็กซ์แพนเดอร์ (Expander) ซึ่งใช้สำหรับด้านรับ ในบริเวณที่มีแอมพลิจูดต่ำๆ จะมีการใช้ควอนไทซ์ซิงที่แคบๆ ส่วนในบริเวณที่มีสัญญาณแอมพลิจูดสูงๆ จะใช้ควอนไทซ์ซิงที่กว้างๆ การแบ่งที่มีระดับไม่เท่ากันนั้น จะเรียกการแปลงสัญญาณแบบนี้ว่า Adaptive Pulse Code Modulation : ADPCM.



(ก) คอมเพรสชัน (ข) ยูนิฟอร์มควอนไทซ์ซิง (ค) นอนยูนิฟอร์มควอนไทซ์ซิง (ง) เอ็กซ์แพนชัน

รูปที่ 2.7 แสดงคุณลักษณะการจัดระดับด้วยคอมเพรสเซอร์และเอ็กซ์แพนเดอร์

## บทที่ 8

### การแปลงฟูรีเยร์

#### (FAST FOURIER TRANSFORM)

##### 8.1 ทฤษฎีการแปลงฟูรีเยร์

อุปกรณ์คอมพิวเตอร์ในปัจจุบันมีราคาถูกและมีประสิทธิภาพสูง อีกทั้งความแม่นยำและความแน่นอนจากการคำนวณโดยใช้คอมพิวเตอร์มีมาก จึงทำให้การวิจัยและพัฒนาของการประมวลสัญญาณเชิงตัวเลข เพื่อนำไปประยุกต์ใช้กับงานในสาขาวิชาต่างๆ โดยเฉพาะทางด้านอิเล็กทรอนิกส์มีอย่างกว้างขวาง ตัวอย่างเช่น การประมวลสัญญาณเสียงเชิงตัวเลข (digital speech processing) การประมวลภาพ (digital image processing) การประมวลสัญญาณเรดาร์ และการประมวลสัญญาณชีวภาพ (digital processing of biomedical signal) เป็นต้น โดยทั่วไปแล้วการแปลงฟูรีเยร์แบบไม่ต่อเนื่อง (DFT) นั้นสามารถนำมาใช้ในการคำนวณการประสานได้ แต่การคำนวณ DFT เองนั้นเมื่อลำดับข้อมูลมีมากก็ใช้เวลาในการคำนวณของคอมพิวเตอร์มากตัวอย่างเช่นการคำนวณ DFT สำหรับสัญญาณยาว  $N$  ลำดับหรือจุดนั้น คอมพิวเตอร์ต้องทำการคูณจำนวนเชิงซ้อน (Complex number) ถึง  $N \times N$  ครั้งและบวกจำนวนเชิงซ้อนอีก  $N(N-1)$  ครั้ง ซึ่งคอมพิวเตอร์ทั่วไปแล้วไม่มีคำสั่งภาษาเครื่องที่ใช้ในการคูณตัวเลข จึงต้องเขียนเป็นโปรแกรมย่อย หรือเพิ่มเติมวงจรคูณโดยเฉพาะเข้าไป ส่วนการบวกตัวเลขของคอมพิวเตอร์นั้นทำได้ง่ายและเร็ว จึงสามารถกล่าวได้ว่ากระบวนการคูณตัวเลขนั้นใช้เวลาคำนวณมากกว่าการบวกตัวเลขมาก จากผลอันนี้ทำให้เห็นชัดว่าความเร็วในการคำนวณ DFT จึงขึ้นอยู่กับความเร็วและจำนวนครั้งในการคูณตัวเลขเป็นสำคัญ ถ้าหากสามารถหาวิธีการคำนวณที่ใช้จำนวนครั้งในการคำนวณน้อยลงแล้วก็จะประโยชน์อย่างมาก

ขั้นตอนวิธีหรือลำดับการในการคำนวณ DFT ให้เร็วที่มีชื่อเรียกว่าการแปลงฟูรีเยร์ หรือ FFT ที่ได้มีการพัฒนาวิธีการต่างๆ กันอย่างมากนั้นเป็นผลมาจากผลงานของ J.W. Cooley กับ J.W. Tukey ที่เสนอไว้ในปี ค.ศ. 1965

FFT จะทำให้การคำนวณ DFT ใช้การคูณจำนวนเชิงซ้อนเพียง  $N \log_2 N$  ครั้งเท่านั้น หรือจำนวนครั้งในการคูณตัวเลขลดลงไปถึง  $N/(\log_2 N)$  เท่า ตัวอย่างเช่น สำหรับลำดับสัญญาณขนาด  $N = 1024$  จุด การคูณจำนวนเชิงซ้อนลดลงไปได้ถึง 102 เท่า ผลคืออีกประการหนึ่งก็คือ ทำให้การสร้างวงจรเฉพาะเพื่อการคำนวณ DFT ทำได้ง่ายและคำนวณได้เร็วขึ้นเป็นผลให้การประมวลสัญญาณเชิงเลขสามารถนำไปประยุกต์ใช้กับระบบเวลาจริงได้ อย่างไรก็ตามถึงแม้ FFT จะมีชื่อเรียกว่า การแปลงฟูรีเยร์ แต่ตัว FFT เองนั้นไม่ใช่การแปลงฟูรีเยร์เป็นเพียง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

วิธีการหรือลำดับการในการคำนวณที่ช่วยให้การคำนวณ DFT ซึ่งเป็นการแปลงฟูเรียร์ได้รวดเร็วขึ้น

จนถึงปัจจุบันนี้ได้มีการคิดค้นค้นคิดแปลง และเสนอผลงานเกี่ยวกับ FFT มากมายหลายแบบ แต่ละแบบก็มีข้อดีข้อเสียต่างกันไป โดยทั่วไปแล้วเราอาจแบ่งออกได้เป็นชนิดใหญ่ ๆ 2 ชนิดด้วยกันคือ ชนิดลดทอนทางเวลา (Decimation in Time) กับชนิดลดทอนทางความถี่ (Decimation in Frequency) ทั้งสองชนิดนี้โดยหลักการแล้วมีความคล้ายคลึงกัน ดังนั้นจึงขออธิบายหลักการแบบลดทอนทางเวลาซึ่งนำมาใช้ในโครงงานนี้อย่างละเอียดเพียงอย่างเดียว เพื่อให้เห็นถึงวิธีการว่า FFT สามารถลดการคำนวณลงได้อย่างไร

### 3.2 หลักการเบื้องต้นของ FFT

การแปลงฟูเรียร์แบบไม่ต่อเนื่องสำหรับลำดับ  $x(m)$  ที่ยาว  $N$  จุดสามารถนิยามได้ดังนี้

$$X(k) = \sum_{m=0}^{N-1} x(m)W^{mk} \quad (3.1)$$

โดยที่ดัชนี  $k = 0, 1, \dots, N-1$  และจำนวนเชิงซ้อน  $W = \exp(-2\pi j/N)$  ในทางวิชาอิเล็กทรอนิกส์ลำดับ  $x(m)$  มักจะเกี่ยวข้องกับสัญญาณในโดเมนเวลา ส่วน  $X(k)$  เกี่ยวข้องกับสัญญาณในโดเมนความถี่ หรือเรียกว่าสเปกตรัมของสัญญาณ

สมการ (3.1) สามารถเขียนให้อยู่ในรูปของสมการเมทริกซ์ได้คือ

$$\{X\} = \{A\}\{x\} \quad (3.2)$$

โดยที่  $X$  และ  $x$  เป็นเวกเตอร์แนวตั้งที่ประกอบด้วยลำดับ  $X(k)$  และ  $x(m)$  ตามลำดับจำนวน  $N$  ลำดับ และ  $A$  เป็นเมทริกซ์จัตุรัส ขนาด  $N \times N$  ที่มีสมาชิกเป็นจำนวนเชิงซ้อน  $w^{mk}$  ตัวอย่างเช่น พิจารณาสำหรับกรณี  $N=4$  ระบบสมการ (3.2) สามารถเขียนแยกได้เป็น

$$\begin{bmatrix} X(0) \\ X(1) \\ X(2) \\ X(3) \end{bmatrix} = \begin{bmatrix} w^0 & w^0 & w^0 & w^0 \\ w^0 & w^1 & w^2 & w^3 \\ w^0 & w^2 & w^4 & w^6 \\ w^0 & w^3 & w^6 & w^9 \end{bmatrix} \begin{bmatrix} x(0) \\ x(1) \\ x(2) \\ x(3) \end{bmatrix} \quad (3.3)$$

การคำนวณจากสมการ (3.3) โดยตรงนั้นสังเกตเห็นได้ว่าต้องใช้การคูณจำนวนเชิงซ้อน  $4 \times 4$  ครั้ง และต้องทำการบวกจำนวนเชิงซ้อน  $N(N-1) = 4(4-1) = 12$  ครั้ง ประโยชน์ด้านการคำนวณการเป็นเอกสารเป็นเอกสารที่สงวนลิขสิทธิ์ไว้เพื่อใช้ในการศึกษาวิจัยและการเผยแพร่โดยไม่หวังกำไร  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ที่นี้มาดูหลักการสำคัญประการหนึ่งของการแปลงฟูรีเยร์ หรือ FFT ที่ลดจำนวนครั้งในการคูณจำนวนเชิงซ้อนได้โดยอาศัยคุณสมบัติภาวะเป็นคาบของจำนวนเชิงซ้อน  $w$  คือ

$$w^{mk} = w [mk \bmod N] \quad (3.4)$$

ซึ่ง  $mk \bmod N$  หมายถึงส่วนที่เหลือจากการหารพจน์  $mk$  ด้วย  $N$  โดยอาศัยคุณสมบัติเป็นคาบ ทำให้สมการ (3.3) อาจเขียนได้เป็น

$$\begin{bmatrix} X(0) \\ X(1) \\ X(2) \\ X(3) \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & w^1 & w^2 & w^3 \\ 1 & w^2 & w^0 & w^2 \\ 1 & w^3 & w^2 & w^1 \end{bmatrix} \begin{bmatrix} x(0) \\ x(1) \\ x(2) \\ x(3) \end{bmatrix} \quad (3.5)$$

จากคุณสมบัติภาวะเป็นคาบนี้เอง ทำให้เราสามารถแยกตัวประกอบของเมทริกซ์  $A$  ออกเป็นเมทริกซ์ย่อยหลายเมทริกซ์คูณกัน และสมาชิกภายในเมทริกซ์ย่อยให้มีค่าเป็นศูนย์มากที่สุด วิธีการแยกตัวประกอบนี้จะไม่กระทำโดยตรงจาก  $A$  แต่จะมีการสลับตำแหน่งหรือจัดกลุ่มของเมทริกซ์(อาจมีการสลับสมาชิกตามแนวตั้ง หรือสมาชิก ตามแนวนอน อย่างใดอย่างหนึ่ง) ด้วยวิธีการของผันกลับบิต (bit reverse) และเมทริกซ์หลังการทำสลับแถวแล้วจะนำมาแยกตัวประกอบอีกครั้งหนึ่ง

### 3.3 ขั้นตอนวิธีลดทอนทางเวลา (Decimation in Time DIT)

วิธีการที่เสนอที่เสนอโดย คูลีย์ และ ทูคีย์ ซึ่งได้กล่าวมาแล้วนั้น ความจริงแล้วเป็นการจัดแบ่งกลุ่มสัญญาณในโดเมนเวลา  $x(m)$  ที่มีขนาด  $N$  จุดออกเป็นสองลำดับสัญญาณที่มีความยาว  $N/2$  จุด เท่ากัน ทั้งสองลำดับนี้ให้ชื่อเรียกว่า ลำดับ สัญญาณคู่ และ ลำดับสัญญาณคี่ โดยที่ลำดับสัญญาณคู่เกิดจากการเอาลำดับในตำแหน่งเป็นเลขคู่มาเรียงกันที่เหลือก็เป็นลำดับสัญญาณคี่ ดังนั้น ถ้าเรานิยามให้  $x_E(m)$  เป็นลำดับคู่  $x_O(m)$  เป็นลำดับคี่ เพราะฉะนั้น

$$x_E(m) = x(2m) \quad ; \quad m = 0, 1, \dots, (N/2)-1$$

(3.6)

$$x_O(m) = x(2m + 1) \quad ; \quad m = 0, 1, \dots, (N/2)-1$$

ด้วยการจัดแบ่งเช่นนี้ ถ้าให้ค่า  $W_N$  แทนค่า  $W$  ของลำดับยาว  $N$  จุด ทำให้การคำนวณ DFT ของลำดับสัญญาณ  $x(n)$  ที่ยาว  $N$  จุดเขียนใหม่ได้เป็น

$$\begin{aligned}
 X(k) &= \sum_{m=0}^{N-1} x_E(m) (W_N)^{km} + \sum_{m=0}^{N-1} x_O(m) (W_N)^{km} \\
 &= \sum_{m=0}^{(N/2)-1} x(2m) (W_N)^{2mk} + \sum_{m=0}^{(N/2)-1} x(2m+1) (W_N)^{(2m+1)k}
 \end{aligned} \tag{3.7}$$

โดยที่เขียนให้พจน์  $(W_N)^2$  เป็น

$$(W_N)^2 = [\exp(j2\pi/N)]^2 = W_{N/2}$$

ซึ่ง  $W_{N/2}$  หรือค่า  $W$  ของลำดับยาว  $N/2$  จุด เพราะฉะนั้น (3.7) จัดพจน์ใหม่ได้คือ

$$X(k) = \sum_{m=0}^{(N/2)-1} x_E(m) (W_{N/2})^{km} + (W_N)^k \sum_{m=0}^{(N/2)-1} x_O(m) (W_{N/2})^{km}$$

$$X(k) = X_1(k) + (W_N)^k X_2(k) \tag{3.8}$$

โดยที่  $X_1(k)$  และ  $X_2(k)$  แทนผลการแปลง DFT ขนาด  $N/2$  จุดของลำดับ  $x_E(m)$  และ  $x_O(m)$  ตามลำดับ สมการที่ (3.8) แสดงให้เห็นว่าการคำนวณ DFT ขนาด  $N$  จุดนั้นสามารถแบ่งคำนวณย่อยออกเป็นการคำนวณ DFT ขนาด  $N/2$  จุดสองอันนับได้ และข้อสำคัญคือ การคูณจำนวนเชิงซ้อนจะลดลงเหลือ  $2(N/2)^2 = N^2/2$  ครั้ง ซึ่งจะเห็นว่าการลดเวลาการคำนวณลงไปได้ถึง 50 เปอร์เซ็นต์ โดยอาศัยหลักการเดียวกันถ้าเราแบ่งทอนลำดับ  $x_E(m)$  และ  $x_O(m)$  ออกเป็นลำดับคู่และลำดับคี่ลงไปอีกตามลำดับ จนในที่สุดเหลือเป็นลำดับขนาดเพียงสองจุดหรือกล่าวได้ว่า การคำนวณการแปลง DFT ขนาด  $N$  จุดทำได้โดยการคำนวณการแปลง DFT ขนาด 2 จุด จำนวน  $N/2$  ภาคด้วยกัน ข้อสังเกตที่สำคัญก็คือการซอยเพื่อแบ่งลำดับ  $x(n)$  ออกเป็น

ทีละครึ่งจนเหลือการคำนวณ DFT ขนาด 2 จุดนี้ สำหรับสัญญาณขนาด  $N$  ลำดับ จะทำการแบ่งออกได้  $\log_2 N$  ครั้ง

ปัญหาต่อมาเมื่อแบ่งย่อยลงไปแล้วจะเอาการแปลง DFT ขนาด 2 จุดจำนวน  $N/2$  นี้มาประกอบกัน หรือรวมตัวกันเพื่อให้เป็นการคำนวณ DFT ขนาด  $N$  ได้อย่างไร เพราะถ้าหากนำมาประกอบกันโดยไม่มีหลักเกณฑ์ ผลการคำนวณ DFT ที่ได้มีค่าผิดไป การนำมาประกอบกันสำหรับลำดับสัญญาณ  $N$  ลำดับอย่างถูกต้อง เราต้องทำการนิยามค่าของสมการ (3.8) สำหรับค่า  $k > N/2$  ค่ะ

$$\begin{aligned} X(k) &= X_1(k) + (W_N)^k X_2(k) \quad ; 0 \leq k \leq (N/2)-1 \\ &= X_1(k-N/2) + (W_N)^k X_2(k-N/2) \quad ; N/2 \leq k \leq N-1 \end{aligned} \quad (3.9)$$

พจน์  $(W_N)^k$  ในสมการ (3.9) นี้มีชื่อเรียกว่า *ตัวประกอบหมุน* พจน์นี้เป็นส่วนสำคัญร่วมกับ DFT ขนาดสองจุด หรือ DFT ขนาด  $N/2$  จุด ที่จะนำประกอบกันเข้าเป็น DFT ขนาด  $N$  จุดเหมือนเดิม

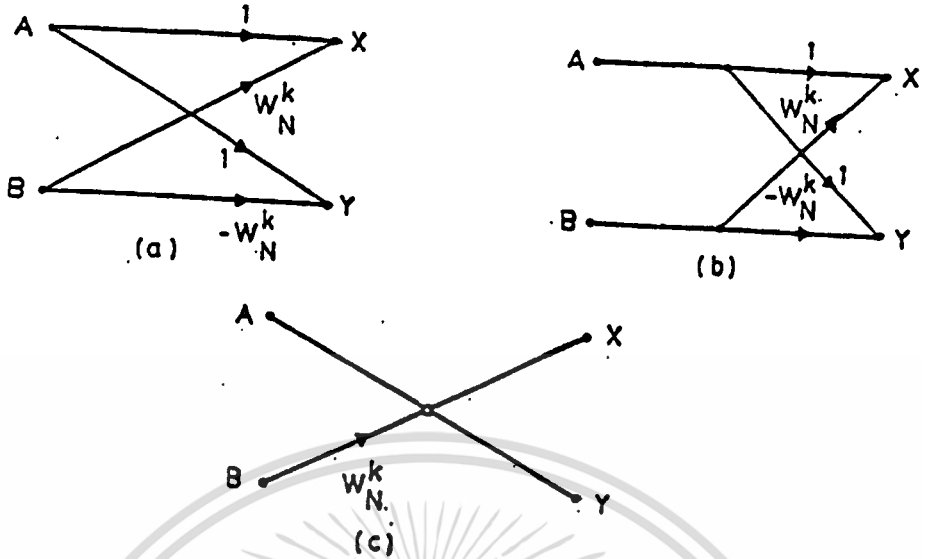
ในสมการที่ (3.9) นี้ ถ้าหากเรานำเอาความสัมพันธ์  $(W_N)^{k-N/2} = -(W_N)^k$  เทนลงไปในสมการนี้เขียนใหม่เป็น

$$X(k) = X_1(k) + (W_N)^k X_2(k) \quad ; 0 \leq k \leq (N/2) \quad (3.10)$$

$$= X_1(k-N/2) + (W_N)^{k-(N/2)} X_2(k-N/2) \quad ; N/2 \leq k \leq (N/2)-1$$

(3.11)

สมการที่ได้นี้เป็นหลักการสำคัญสำหรับ FFT แบบนี้



รูปที่ 3.1 หน่วยผีเสื้อของการคำนวณค่าบนคอนวูลชันทอนทางเวลา

จากสมการบอกเราว่าในการคำนวณ DFT ของลำดับคู่หนึ่ง จะประกอบด้วยลำดับ  $X(k)$  ในสมการ (3.10) และลำดับ  $X(k)$  ใน (3.11) ซึ่งห่างออกไปจากลำดับ  $X(k)$  ใน (3.10) ไป  $N/2$  จุด สามารถคำนวณได้โดยการใช้การคูณจำนวนเชิงซ้อนเพียงครั้งเดียวเท่านั้น จากผลนี้ เราสามารถนำไปสร้างหน่วยคำนวณที่มีชื่อเรียกว่า หน่วยผีเสื้อ (butterfly unit) โดยหน่วยข้อมูลนี้มีข้อมูลเข้าสองข้อมูลคือ A และ B และให้ข้อมูลออกเป็น X และ Y และ

$$X = A + (W_N)^k B$$

(3.12)

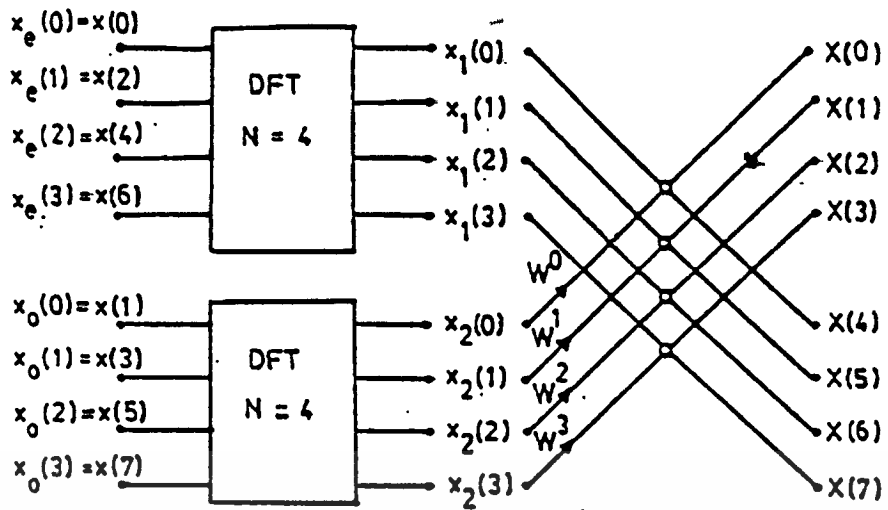
$$Y = A - (W_N)^k B$$

โดยที่การทำงานของหน่วยผีเสื้อนี้ สามารถเขียนแทนด้วยกราฟสัญญาณการไหลดังรูปที่ 3.1(a) 3.1(b) หรือ 3.1(c)

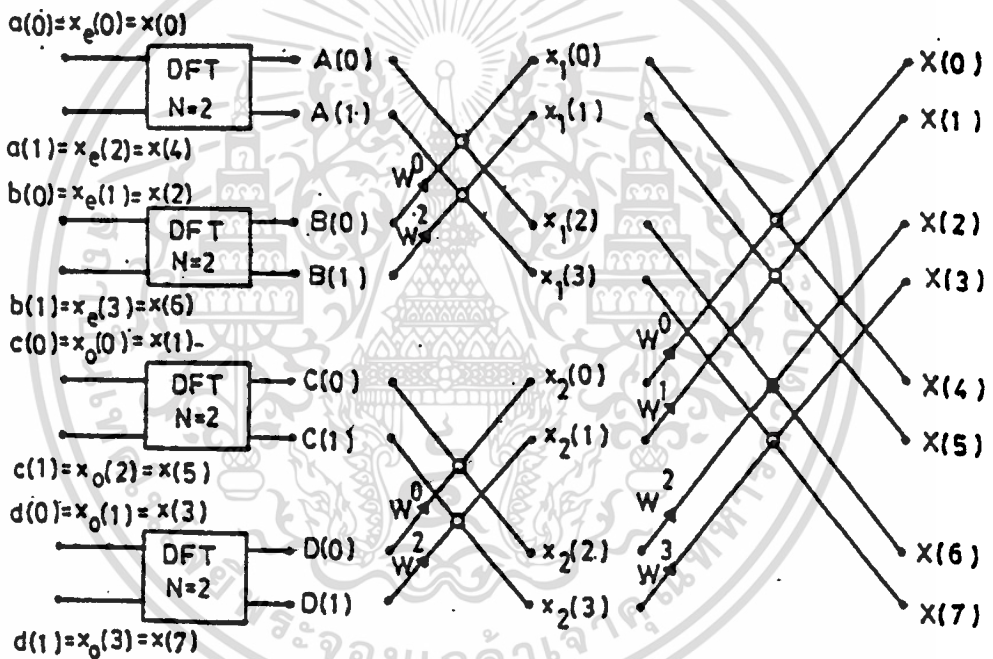
ต่อไปเป็นตัวอย่างการคำนวณ DFT แบบ 8 จุด โดยอาศัยวิธีที่ได้อธิบายมาแล้ว ขั้นตอนการคำนวณได้แสดงอย่างเป็นลำดับไว้ด้วยกราฟการไหลของสัญญาณ ดังรูปที่ 3.2 โดยในเบื้องต้นลำดับของข้อมูลเข้า  $x(n)$  จะถูกแบ่งออกเป็นลำดับคู่  $x_E(m)$  และลำดับคี่  $x_O(m)$  อย่างละสี่ลำดับ ดังรูป 3.2(a) ค่อยจากนั้นลำดับ  $x_E(m)$  และ  $x_O(m)$  ก็ถูกแบ่งย่อยลงไปอีก ดังรูป 3.2(b) ซึ่งในที่สุดการคำนวณ DFT ขนาด 8 จุดถูกสับย่อยออกเป็นการคำนวณ DFT ขนาด 2 จุดสี่ภาคด้วยกัน ดังปรากฏในรูป 3.3(a) ซึ่งเป็นผังการไหลของสัญญาณสำหรับการคำนวณ DFT ขนาด 8 จุด

อย่างสมบูรณ์ สังเกตเห็นว่าในกรณีนี้  $N = 8$  ทำให้  $\log_2 8 = 3$  การคำนวณจึงแบ่งออกเป็น 3 ภาคน้อย แต่แต่ละภาคประกอบด้วยหน่วยสี่เหลี่ยม  $N/2 = 4$  หน่วยและหน่วยสี่เหลี่ยมแต่ละหน่วยใช้การคูณเลขเชิงซ้อนหนึ่งครั้ง เพราะฉะนั้นการคำนวณทั้งหมดมีการคูณตัวเลขเท่ากับ  $8/2 \log_2 8 = 12$

ตามรูป 3.3(a) มีข้อนำสังเกตก็คือ ลำดับสัญญาณเข้า  $x(n)$  ไม่ได้ถูกจัดเรียงอย่างค่อนเนื่องแต่ได้ถูกสลับอันดับอย่างมีหลักเกณฑ์ คือการสลับอันดับนี้จะนำไปตามวิธีการที่เรียกว่า *การผันกลับบิต* (bit reverse) นั่นคือถ้าเราแทนตรรกะ  $n$  ของลำดับ  $x(n)$  ด้วยเลขฐานสองโดยที่จำนวนบิตของเลขฐานสอง 3 บิต จากนั้นการจัดลำดับ  $x(n)$  ใหม่จะได้จากการผันกลับของเลขไบนารีแทนตรรกะ  $n$  ดังแสดงในรูป 3.3(b) คือ  $x(011)$  จะถูกแทนที่ด้วย  $x(100)$  หรือ  $x(110)$  ถูกแทนที่ด้วย  $x(011)$  เป็นต้น จากที่กล่าวมาแล้วว่าปกติตรรกะ  $n$  มักเกี่ยวข้องกับเวลาและวิธีการของ FFT แบบนี้เป็นการลดเวลาการคำนวณโดยการสับหรือตัดทอนลำดับโคเมนในเวลาหรือ  $x(n)$  ออกเป็นกลุ่มย่อยแต่ละกลุ่มประกอบด้วยลำดับ  $x(n)$  เพียงสองลำดับ ซึ่งสองลำดับนี้เป็นบัพคู่กัน การจัดกลุ่มนี้คล้ายคลึงกับการสุ่มตัวอย่างลำดับเดิมอีกครั้งด้วยอัตราการสุ่มตัวอย่างที่ต่ำกว่า และถ้าหากเราถือว่าแต่ละกลุ่มลำดับข้อมูลใหม่ที่จัดขึ้นมารวมเป็นข้อมูลหนึ่งลำดับแล้ว ก็เท่ากับว่าเราได้ตัดทอนลำดับในโคเมนเวลาลงไปเป็นลำดับข้อมูลย่อยหลายลำดับ ดังนั้นการคำนวณแบบนี้จึงมีชื่อเรียกว่า *การลดทอนทางเวลา*

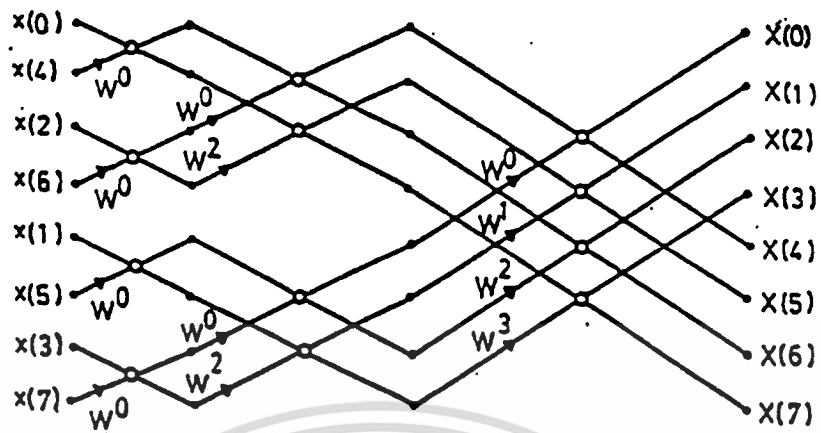


(a)

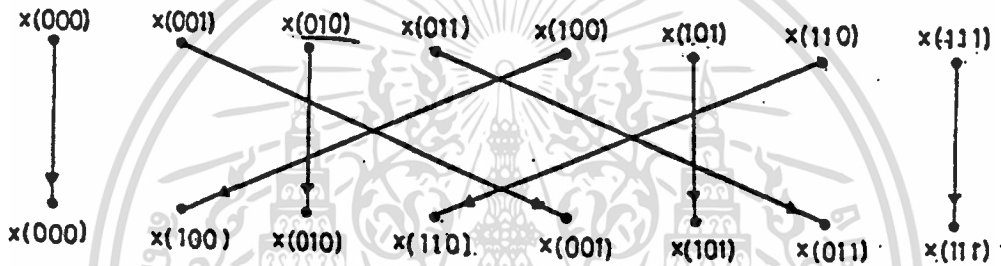


(b)

รูปที่ 3.2(a) และ (b) แสดงขั้นตอนวิธีแบบ DIT สำหรับ DFT แบบ 8 จุด



(a)



(b)

รูปที่ 3.3 (a) กราฟการไหลของสัญญาณแสดงการคำนวณตามรูปที่ 3.2  
(b) แสดงการสลับตำแหน่งของลำดับ  $x(n)$  ด้วยการผันกลับบิต

## บทที่ 4

### การจดจำเสียงพูด

(Speech Recognition)

#### 4.1 ระบบอินพุตเสียงพูด (Speech Input System)

อุปกรณ์อินเทอร์เน็ตที่ใช้มักเป็นคีย์บอร์ดทางกล เช่น คีย์บอร์ด เครื่องอ่านเทปกระดาษ เป็นต้น แต่ยังมีวิธีการติดต่อลักษณะอื่นอีกนอกเหนือไปจากอินพุตทางกลดังกล่าวคือ อินพุตที่เป็นเสียงพูด อุปกรณ์อินพุตที่เป็นเสียงพูดจะดึงข้อมูลข่าวสารและทำหน้าที่แปลความหมายจากสัญญาณเสียงที่เข้ามา ประโยชน์ของอุปกรณ์ดังกล่าวคือ

1) ความเร็วในการอินพุตสูง บางการทดลองเปิดเผยว่า อัตราเร็วของข่าวสารที่อินพุตโดยเสียงพูดมีอัตราเร็วเป็นสามเท่าของการใช้คีย์บอร์ด และเป็นแปดเท่าของการป้อนหน่วยเสียงโดยฝีมือ

2) ไม่จำเป็นต้องมีการฝึกอบรม ซึ่งต้นกำเนิดเสียงมาจากการพูดตามธรรมชาติของมนุษย์

3) สามารถทำการประมวลผลแบบขนาน ไปได้กับการประมวลผลข่าวสารจากแหล่งอื่น

4) ตัวเซ็นเซอร์ของอินพุตสร้างได้ง่ายและประหยัด

เมื่อรวมอุปกรณ์สังเคราะห์เสียงพูด อุปกรณ์จดจำเสียงพูดกับอินพุตเสียงพูดสามารถทำงานได้ในโหมด online พร้อมกับ ตอบสนองกลับทันทีเพื่อยืนยันผลการจดจำและแสดงคำสั่งสำหรับอินพุตถัดไป อินพุตเสียงพูดมีความสะดวกสบายต่อการใช้และมีความเชื่อถือได้

เทคโนโลยีพื้นฐานสำหรับการสร้างระบบอินพุตเสียงพูดนี้ คือการทำอัลกอริทึมการจดจำเสียงพูด อย่างไรก็ตามยังเป็นไปไม่ได้ที่อินพุตเสียงพูดดังกล่าวจะจดจำมนุษย์คนหนึ่งๆ มนุษย์เราสามารถจดจำภาษาสนทนาที่ต่อเนื่องที่มีศัพท์มากมาย แม้ว่าการจดจำเสียงพูดในปัจจุบันยังไม่มี ความสามารถเทียบเท่า แต่จุดมุ่งหมายคือการจดจำเสียงพูดดังกล่าวมีความยากในการแปลงเสียงพูดที่ต่อเนื่องกันหรือภาษาสนทนาให้เป็นลำดับของหน่วยเสียง หรือส่วนน้อยที่สุดของเสียงในภาษา (phoneme) เพราะเหตุว่าลักษณะทางอคูสติกของโฟเนม แปลไปในช่วงกว้างและยังซับซ้อนด้วยอิทธิพลของฉันทลักษณ์ เช่น การออกเสียงสูงต่ำและสำเนียง ปัจจุบันนี้สามารถกล่าวได้ว่า จดจำเสียงพูดได้เมื่อมีการออกเสียงทีละคำแยกกัน โดยมีโครงสร้างของประโยคตามหลักไวยากรณ์ และศัพท์ที่จำกัด

เราสามารถแบ่งวิธีการจดจำเสียงพูดออกได้เป็น 2 วิธี คือ

1. แบบพิจารณาทั้งหน่วยคำที่เปล่งออกมา (Isolated Words Recognition)

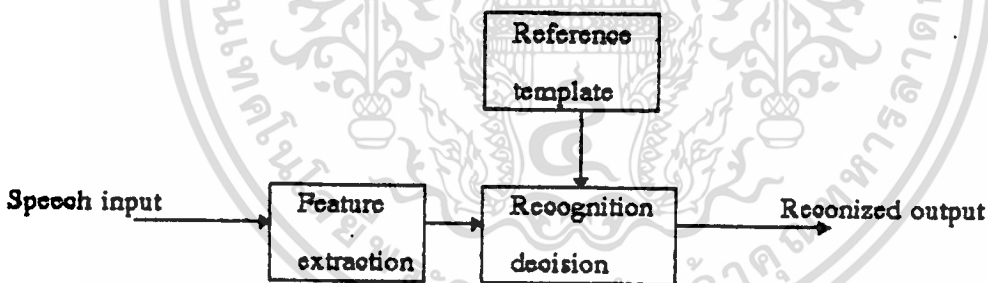
หน่วยคำที่พิจารณาในวิธีนี้ อาจเป็นพยางค์เดี่ยว คำ กลุ่มคำ วลี หรือประโยคก็ได้สำหรับการวิจัยนี้จะเป็นการพิจารณาคำเดี่ยว (Word Based Recognition)

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

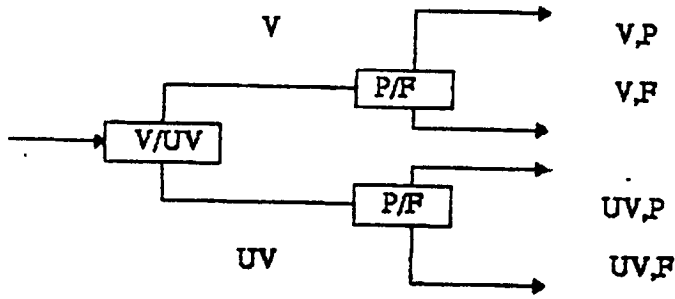
2. การจดจำโดยแยกแยะรายละเอียดของหน่วยเสียง (Phoneme Based Recognition) จะพิจารณาลักษณะของหน่วยเสียงที่มีขนาดเล็กลงไป เช่น ระดับสระ พยัญชนะ หรือวรรณยุกต์ โดยจะใช้หน่วยเสียงย่อยเหล่านี้เป็นหลักในการเปรียบเทียบ วิธีนี้จะสะดวกสำหรับระบบที่จะรับรู้คำจำนวนมาก โดยจะใช้เป็นพื้นฐานในการวิเคราะห์ในขอบเขตที่กว้างขึ้น

#### 4.2 อัดกอลิเทียมและสถาปัตยกรรมของการจดจำเสียงพูด

การจดจำเสียงพูดเป็นกระบวนการซึ่งมีการวิเคราะห์อкуตติค มีการดึงลักษณะทางภาษา แล้วแปลงเป็นสัญลักษณ์ทางภาษา ตามกระบวนการนี้เรามาศึกษาอุปกรณ์สองประเภท ประเภทแรกตามรูปที่ 4.1 ส่วนตัดสินใจหาค่าความเหมือนกันโดยเปรียบเทียบรูปแบบอินพุทของลักษณะหน่วยเสียงกับรูปแบบอ้างอิงซึ่งมีเก็บอยู่ก่อนแล้วจะกระทำหลังการวิเคราะห์ทางอкуตติค อีกประเภทหนึ่งเป็นกระบวนการทำลำดับจากการตัดสินใจค่าเลขไบนารี โดยใช้การวิเคราะห์ทางอкуตติคสำหรับพื้นเสียงของหน่วยเสียงหนึ่งแสดงตามรูป 4.2 โดยไม่ใช้รูปแบบอ้างอิง ในตัวอย่างนี้การตัดสินใจค่าไบนารีมีขึ้นสำหรับลักษณะที่เชื่อมกันของหน่วยเสียงหนึ่งโดยเป็นลำดับขั้น ทั้งสองกระบวนการที่กล่าวถึง แบบแรกที่ใช้รูปแบบอ้างอิงเป็นที่นิยมกว่า แบบที่สองไม่เป็นที่แพร่หลายและจะไม่กล่าวถึงในโครงการนี้



รูปที่ 4.1 กระบวนการจดจำเสียงพูดที่ใช้รูปแบบอ้างอิง



รูปที่ 4.2 กระบวนการจดจำเสียงพูดโดยการตัดสินใจแยกเป็นลำดับ

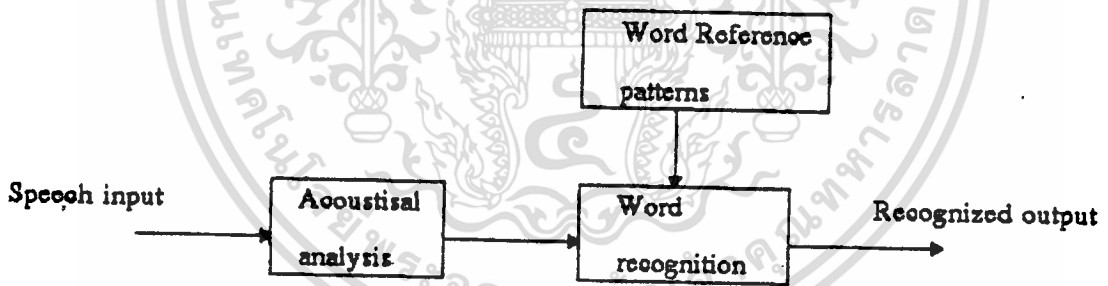
หมายเหตุ V หมายถึง voiced

UV หมายถึง unvoiced

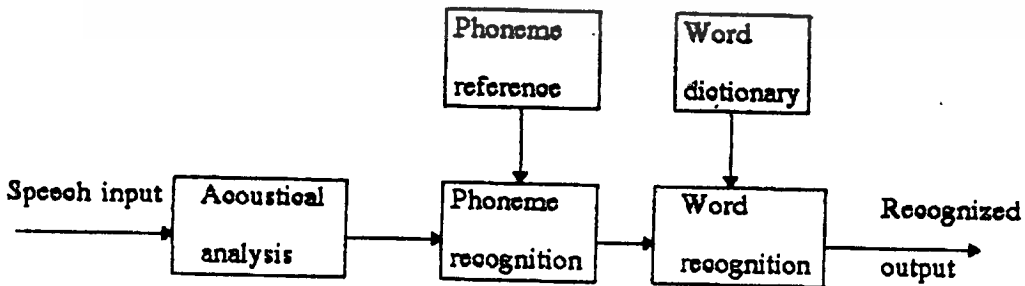
P หมายถึง plosive

และ F หมายถึง fricative

ตัวอย่างอย่างง่ายของรูปที่ 4.1 คือ การจดจำเสียงพูดที่ใช้รูปแบบอ้างอิงของคำที่ซึ่งถือเป็นหน่วยของการวิเคราะห์และการจดจำทางอรรถศาสตร์ที่เรียกว่าระบบจดจำคำ ดังรูป 4.3 ซึ่งเป็นประเภทที่ใช้ในเชิงพาณิชย์ จากการทำผิดพลาดบ่อยครั้งเช่นนี้ทำให้จำนวนของรูปแบบอ้างอิงจะเพิ่มเป็นสัดส่วนกับคำศัพท์ที่ต้องรู้จัก ถ้าคำหนึ่งถูกแทนด้วยลำดับของหน่วยเสียง



รูปที่ 4.3 กระบวนการจดจำคำที่มีรูปแบบอ้างอิงเป็นคำ



รูปที่ 4.4 กระบวนการจดจำคำซึ่งมีรูปแบบอ้างอิงเป็นหน่วยเสียง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สถาปัตยกรรมตามรูปที่ 4.4 สามารถสร้างขึ้นโดยใช้ความจุของหน่วยความจำน้อยกว่ารูปที่ 4.3 ประเภทนี้มีการเก็บรูปแบบอ้างอิงเป็นหน่วยหน่วยเสียงต้องมีปทานุกรมของคำอีกด้วย ปทานุกรมของคำที่ว่าจะอธิบายคำเป็นลำดับของหน่วยเสียงและใช้ความจุหน่วยความจำน้อยกว่าระบบ ในรูปที่ 4.3 มาก ฉะนั้นเมื่อคำศัพท์มีขนาดคำสั้น ระบบตามรูปที่ 4.3 ทำงานได้ดีกว่า ขณะที่รูปที่ 4.4 ทำงานได้ดีกว่าเมื่อขนาดคำศัพท์นั้นยาวขึ้น ในหัวข้อต่อไปจะกล่าวถึงอัลกอริทึมของรูปที่ 4.3 และรูปที่ 4.4

### 1. การวิเคราะห์ทางอคูสติก (Acoustical Analysis)

โดยปรกติข้อมูลข่าวสารสองชนิดคือ เอนเวลโลปของสเปกตรัมเสียงพูดและคุณลักษณะของแหล่งกระตุ้นถูกใช้เป็นพารามิเตอร์สำหรับการจดจำเสียงพูด เอนเวลโลปของสเปกตรัมความถี่จะถูกฟิลเตอร์ประมาณ 10 ชุดดึงออกมา โดยช่วงของฟิลเตอร์ความถี่นี้ควบคุมสเปกตรัมความถี่ของเสียงพูด หรือโดยการประมาณ maximum - likelihood ของสเปกตรัมหรือการวิเคราะห์แบบ

(PARCOR)

การวิเคราะห์แบบฟิลเตอร์ เอาท์พุทของแมนด์พาสฟิลเตอร์ตัวที่  $n$  ใดๆ ส่วน  $A_n(t)$  ประกอบด้วยส่วนย่อยที่แทนด้วยเวกเตอร์ของสเปกตรัมเสียงพูด

$$A(t) = A_1(t), A_2(t), \dots, A_n(t)$$

### 2. รูปแบบการอ้างอิง (Reference pattern)

รูปแบบการอ้างอิงได้มาจากการวิเคราะห์ทางอคูสติกสำหรับคำหรือหน่วยเสียงที่เรารู้ ส่วนประกอบทางภาษาของมัน ในรูปที่ 4.3 รูปแบบการอ้างอิงของแต่ละคำแทนโดยอนุกรมเวลาของลักษณะซึ่งถูกวิเคราะห์ทางอคูสติกแล้ว สำหรับช่วงเวลาเต็มของแต่ละคำ

รูปแบบ อ้างอิงมักประกอบด้วยอนุกรมของเวกเตอร์ หลังจากสุ่มจากเอาท์พุทของฟิลเตอร์สำหรับรูปแบบอ้างอิงทางหน่วยเสียงของรูปที่ 4.4 มักใช้ค่าเฉลี่ยของลักษณะทาง phonemic ที่ถูกวิเคราะห์ในช่วงเวลาของหน่วยเสียง

ตัวอย่างของเสียงพูดที่จะใช้ทำรูปแบบอ้างอิงของหน่วยเสียง เลือกมาจากเสียงพูดที่ต้องจดจำหน่วยเสียงถูกแยกส่วนมาจากคำและถูกเฉลี่ยเพื่อสร้างรูปแบบอ้างอิง สำหรับบางหน่วยเสียงลักษณะทางอคูสติกแปรค่า และยากที่จะตัดสินว่าจะเลือกช่วงเวลาอย่างไร ในกรณีนี้ จะมีการเพิ่มเป็นหน่วยเสียงอ้างอิงพิเศษ สมมติ  $\epsilon$  เท่ากับ 10 และ 16 บิตในทุกๆ คำของ  $A_n$  ความจุหน่วยความจำที่ใช้สำหรับหน่วยเสียงของภาษาญี่ปุ่นแบบเดิมๆ คือเป็นหลัก 10 กิโลบิต สำหรับภาษาอังกฤษจะน้อยกว่านี้เล็กน้อย สำหรับ 10 ดิจิต ความจุหน่วยความจำของรูปแบบอ้างอิงของคำก็จะมีค่ามากกว่านี้ ดังนั้นสำหรับศัพท์ที่มีขนาดยาวมักใช้รูปแบบอ้างอิงของหน่วยเสียง

การวิเคราะห์ทางอคูสติก ค จะดึงเอาคุณลักษณะของแหล่งกำเนิดรวมทั้งข้อมูลว่าสารของคาบเวลาสั้นๆและขนาดของแหล่งกำเนิด ลักษณะเหล่านี้ใช้เมื่อต้องการแบ่งหน่วยเสียงอย่างหยาบๆในการเตรียมเข้าสู่กระบวนการ เช่น การตรวจจับช่วงเวลาของเสียงพูด และการแยกความแตกต่างของ voice และ unvoice

ในการใช้รูปแบบอ้างอิงของคำหรือหน่วยเสียง เราจะได้อัตราการจดจำที่สูงเมื่อใช้ผู้พูดเฉพาะผู้พูดหนึ่ง แต่คำนี้จะลดลงเมื่อใช้เสียงจากผู้พูดอื่น การที่มันขึ้นอยู่กับแต่ละบุคคล โดยพยายามใช้รูปแบบอ้างอิงโดยเฉลี่ย ซึ่งมาจากผลการวิเคราะห์ผู้พูดทั้งหลาย โดยเป็นค่าเฉลี่ยทางอคูสติก แต่กระบวนการเฉลี่ยธรรมดาไม่สามารถครอบคลุมแต่ละคำและลักษณะทางอคูสติกของผู้พูดหลายคน ก็แตกต่างไปจากค่าเฉลี่ยจึงต้องมีการพัฒนาอัตราการจดจำ ถ้าสามารถใช้รูปแบบที่ถูกวิเคราะห์ของผู้พูดเป็นรูปแบบอ้างอิงอัตราของการจดจำจะสูง แต่ความจุหน่วยความจำของรูปแบบอ้างอิงและเวลาในการประมวลผลสำหรับการจดจำ จะกลายเป็นข้อจำกัดในทางปฏิบัติ ดังนั้นเพื่อพัฒนาอัตราการจดจำสำหรับผู้พูดใดๆ จึงมีการดึงเอารูปแบบอ้างอิงสำหรับผู้พูดแต่ละคนหรือสำหรับกลุ่มเล็กๆที่แบ่งผู้พูดออกตามการวิเคราะห์ทางอคูสติกของคำหรือหน่วยเสียง

ขบวนการดึงเอารูปแบบอ้างอิงสำหรับผู้พูดหนึ่งๆเรียกว่า การอบรมหรือการเรียนรู้ของรูปแบบอ้างอิง ระบบจดจำเสียงพูดที่ใช้กระบวนการอบรมของรูปแบบอ้างอิง จะใช้สำหรับผู้พูดที่มีการเรียนรู้โดยสมบูรณ์แล้ว และจะใช้กับผู้พูดที่ไม่ได้ฝึกอบรมไม่ได้ เราเรียกระบบนี้ว่า ระบบจดจำเสียงพูดหรืออุปกรณ์จดจำเสียงพูด สำหรับผู้พูดเฉพาะราย อัตราการจดจำจะขึ้นถึงระดับสำคัญในทางปฏิบัติ ซึ่งใช้ในกระบวนการอบรม แต่การอบรมเองจะใช้เวลานาน

ในกรณีของรูปแบบอ้างอิงที่เป็นคำ ทุกๆคำที่ใช้จะต้องถูกออกเสียงอย่างน้อยหนึ่งครั้งสำหรับการอบรม แม้แต่กับรูปแบบอ้างอิงที่เป็นหน่วยเสียง หน่วยเสียงที่ถูกแยกจากหนึ่งคำในคำศัพท์ จะใช้อัตราการจดจำที่ต่ำกว่าหน่วยเสียงจากหน่วยเสียงเดี่ยวๆธรรมดา และการออกเสียงของคำก็จำเป็นในการอบรมเหมือนกับกรณีรูปแบบอ้างอิงที่เป็นคำ

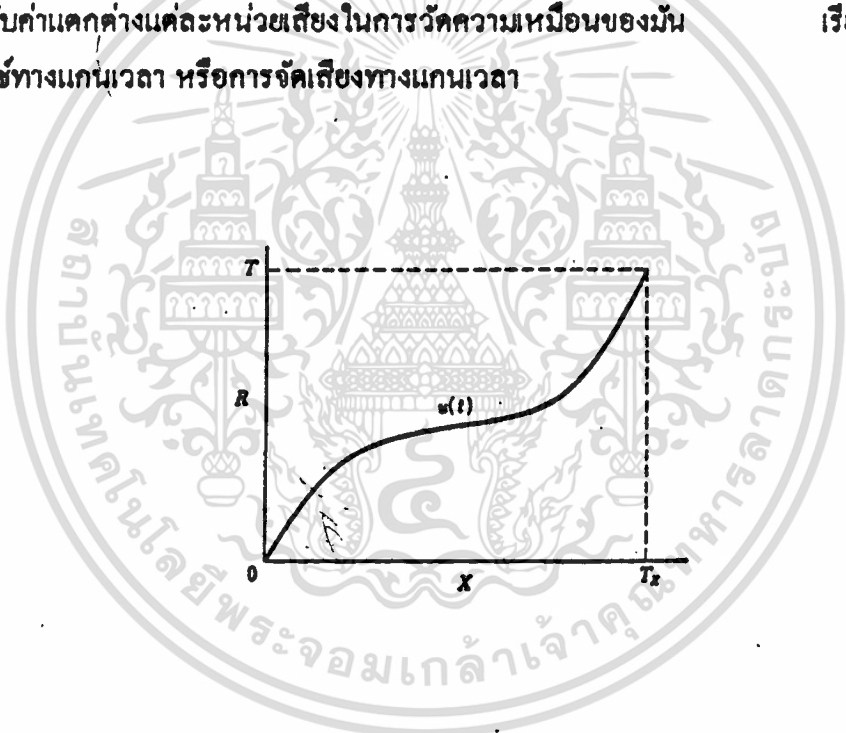
เพื่อให้กระบวนการฝึกอบรมง่ายขึ้นจะต้องมีการศึกษากระบวนการการเรียนรู้และวิธีพื้นฐานที่มีประสิทธิภาพสำหรับรูปแบบอ้างอิง การแบ่งรูปแบบอ้างอิงเป็นกลุ่มย่อยตามอายุและเพศของผู้พูดหรือหน้าที่ในการตัดสินใจที่เป็นแบบไม่ต่อเนื่อง หาได้จากข้อมูลการวิเคราะห์ทางอคูสติกของผู้พูดทั้งหลาย

ระบบจดจำเสียงพูดที่ไม่ต้องใช้กระบวนการเรียนรู้เรียกว่า ระบบจดจำเสียงพูดสำหรับผู้พูดไม่เฉพาะเจาะจง ในระบบนี้มีศัพท์จำนวนมากโดยใช้ระดับของอัตราการจดจำเดียวกันกับของที่เฉพาะเจาะจงผู้พูดประมาณจำนวนหลัก 10 คำ แต่ละคำนั้นมีความแตกต่างกันทาง

อคูสติค รวมทั้งเมื่อขนาดของศัพท์ยาวขึ้นระดับการจดจำก็ลดลงด้วย รูปคลื่นที่เข้ามาในระบบแปรตามไมโครโฟนที่ใช้ ระดับเสียงหรือคุณลักษณะทางอคูสติคของห้องหรือสภาพแวดล้อมอิทธิพลเหล่านี้เป็นปัญหาที่ค้องหาทางแก้ไขต่อไป

ร. การจดจำเป็นหน่วยเสียงและคำโดยใช้รูปแบบอ้างอิง กระบวนการจดจำโดยใช้รูปแบบอ้างอิงเป็นค่าแตกต่างกับรูปแบบอ้างอิงแบบหน่วยเสียง เพราะว่ารูปแบบอ้างอิงของคำถูกแทนโดยเวกเตอร์ของลักษณะ ซึ่งมีอนุกรมเวลาเป็นรูปแบบของมัลติเฟรม โดยขณะที่รูปแบบอ้างอิงของหน่วยเสียงเป็นเฟรมเดียวซึ่งประกอบขึ้นด้วยค่าเฉลี่ยของลักษณะในช่วงหน่วยเสียงนั้น

ในการออกเสียงคำ ช่วงเวลาระหว่างหน่วยเสียงแต่ละคำจะแปรไปบนช่วงกว้างที่สัมพันธ์กัน ดังนั้นจึงจำเป็นที่ค้องหาความเข้ากันได้ระหว่างรูปแบบอ้างอิง และรูปแบบอินพุท โดยการปรับค่าแตกต่างแต่ละหน่วยเสียงในการวัดความเหมือนของมัน เรียกว่าเป็นการนอร์มัลไลซ์ทางแกนเวลา หรือการจัดเสียงทางแกนเวลา



รูปที่ 4.5 การนอร์มัลไลซ์ทางแกนเวลาโดย mapping function  $u(t)$

#### 4. การจดจำคำโดยใช้พทานุกรมของคำ (Word Dictionary).

พทานุกรมของคำมีลำดับของหน่วยเสียงในลำดับของมัน ตัวอย่างเช่น คีจิค "2" ในภาษาญี่ปุ่นคือ /ni/ และในภาษาอังกฤษคือ /tu/ สำหรับหน่วยเสียงที่เหมือนกันช่วงที่ยอมได้ของระยะเวลาระหว่างสั้นที่สุดกับนานที่สุดถูกกำหนดคของคำนั้นๆ ความคล้ายคลึงระหว่างลำดับของหน่วยเสียงของคำและอินพุทเสียงพูดจะถูกตัดสินโดยพทานุกรมศัพท์

## 5. การจดจำที่ต่อเนื่อง (Continuous Word Recognition)

การจดจำโดยใช้รูปแบบอ้างอิงเป็นคำหรือหน่วยเสียง ใช้ข้อมูลข่าวสารที่เป็นช่วงความยาวนานของอินพุตเสียงพูด และสมมติว่ามีเพียงแค่หนึ่งคำในช่วงเวลานั้น อย่างไรก็ตามการพัฒนาการจดจำเสียงพูดที่ต่อเนื่องสำหรับจิต ในกรณีของข้อมูลอินพุตทางตัวเลขและเสียงเป็นที่ต้องการอย่างมากอัลกอริทึมที่ต้องการเปลี่ยนแปลงสำหรับการจดจำเสียงพูดที่ต่อเนื่อง โดยใช้เพียงรูปแบบอ้างอิงเป็นคำเดี่ยวๆ ได้ถูกเสนอแตกต่างกัน

## 6. การเข้าใจในเสียงพูด (Speech Understanding)

สำหรับการจดจำเสียงพูดต่อเนื่องโดยทั่วไปจำเป็นต้องใช้ข้อมูลทาง syntax และ semantic เพื่อทำข้อผิดพลาดทางอรรถศาสตร์ให้ถูกต้อง ในรูปที่ 4.5 กระบวนการทางภาษาจะจัดการหลังการวิเคราะห์ทางอรรถศาสตร์ เรียกว่าวิธีการ bottom up เมื่อหลังการวิเคราะห์ทางภาษาแล้ว อาจมีการวิเคราะห์ทางอรรถศาสตร์ขึ้นกับผลการวิเคราะห์นั้น เรียกว่าวิธีการ top down ซึ่งจะวิเคราะห์ทางอรรถศาสตร์เมื่อจำเป็น นอกจากนี้อาจวิเคราะห์ทางภาษาและทางอรรถศาสตร์ไปพร้อมๆกัน แต่วิธีที่เป็นธรรมชาติกว่าคือ กระบวนการ bottom up

การคิดทิศทางในการวิเคราะห์ทางภาษา เป็นไปได้หลายทางที่หลักๆคือ

1. จากซ้ายไปขวา จะเริ่มจากซ้ายของประโยคมาทางด้านขวา
2. ใช้ key word คือหา key word ของประโยคที่เรียกว่า Island แล้วจึงต่อไปทางซ้ายหรือขวา
3. วิธี left hybrid คือเริ่มจากที่เป็น Island จากนั้นไล่ไปทางซ้ายจนถึงต้นประโยคและย้ายมาทางด้านขวาจนถึงท้ายประโยค

ในจำนวนวิธีทั้งหมด วิธีจากซ้ายไปขวา เป็นพื้นฐานที่สุด และเป็นทิศทางที่เป็นธรรมชาติที่สุด

## 4.3 ความแยกประเภทและตัวอย่างของระบบการจดจำคำ

ระบบการจดจำคำมีสถาปัตยกรรมและฟังก์ชันการทำงานมากมายหลายชนิด และแบ่งได้ตามสิ่งต่างๆต่อไปนี้

1. ผู้พูด คือ เป็นแบบเฉพาะเจาะจง หรือไม่เจาะจงผู้พูด
2. รูปแบบอ้างอิง เป็นคำหรือส่วนน้อยที่สุดของเสียง
3. โหมดการออกเสียงของอินพุตเสียงพูด เป็นแบบคำเดี่ยวหรือต่อเนื่อง ในการจดจำให้เป็นจริงขึ้น โดยรวมเอาสิ่งเหล่านี้ ซึ่งได้อธิบายในที่นี้ด้วย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 1. ผู้พูดเฉพาะเจาะจง และประเภทรูปแบบการอ้างอิงเป็นคำ

อุปกรณ์จดจำเสียงพูดที่ใช้ในเชิงพาณิชย์เป็นประเภทใช้คำเป็นรูปแบบอ้างอิง ซึ่งต้องอาศัยกระบวนการฝึกอบรมสำหรับทุกคำที่ให้มาสำหรับผู้ใช้แต่ละราย ใช้งานได้ทั้งโหมดการพูดที่ต่อเนื่อง และโหมดคำพูดเป็นคำเดียว

ชุดของแบบคัพท์วาทศรในรูปแบบอนาล็อก หรือดิจิทัล ถูกใช้เป็นวิธีวิเคราะห์ทางอรรถศิกในทั้งสองโมเดล โมเดลแบบแรก(Model 580) สามารถรายงานออกมาได้ว่าจดจำคำศัพท์ได้มากที่สุด 256 คำ โดยมีอัตราจดจำมากกว่า 99เปอร์เซ็นต์ ภายหลังมีการเรียนรู้ 10 ครั้งต่อ 1 คำ โมเดลแบบหลังใช้อัลกอริทึมสำหรับการจดจำที่ต่อเนื่องและมากที่สุดได้ 120 คำ มีอัตราการจดจำ 99.3% สำหรับคำเดี่ยวๆ และ 99.6เปอร์เซ็นต์ สำหรับคำที่มี 2-4 ดิจิตโดยเฉลี่ย

### 2. ผู้พูดเฉพาะเจาะจงและรูปแบบการอ้างอิงเป็นหน่วยเสียง

ตัวอย่างการใช้รูปแบบการอ้างอิงเป็นหน่วยเสียงสำหรับผู้พูดที่เฉพาะเจาะจง คือระบบการจดจำคำของ Musashino Electrical Communication Research Laboratory ของ NNT และมหาวิทยาลัยเกียวโตของญี่ปุ่น

การทดลองอินพุตเสียงพูดสำหรับการจองเที่ยวบินภายในประเทศ มีการรายงานผลการจดจำเกือบ 100% ระบบที่ใช้ในการทดลองถูกพัฒนาที่ Musashino ในรูปแบบการเชื่อมต่อคำถามคำตอบกับผลตอบสนองเสียงพูดจากการสังเคราะห์เสียง

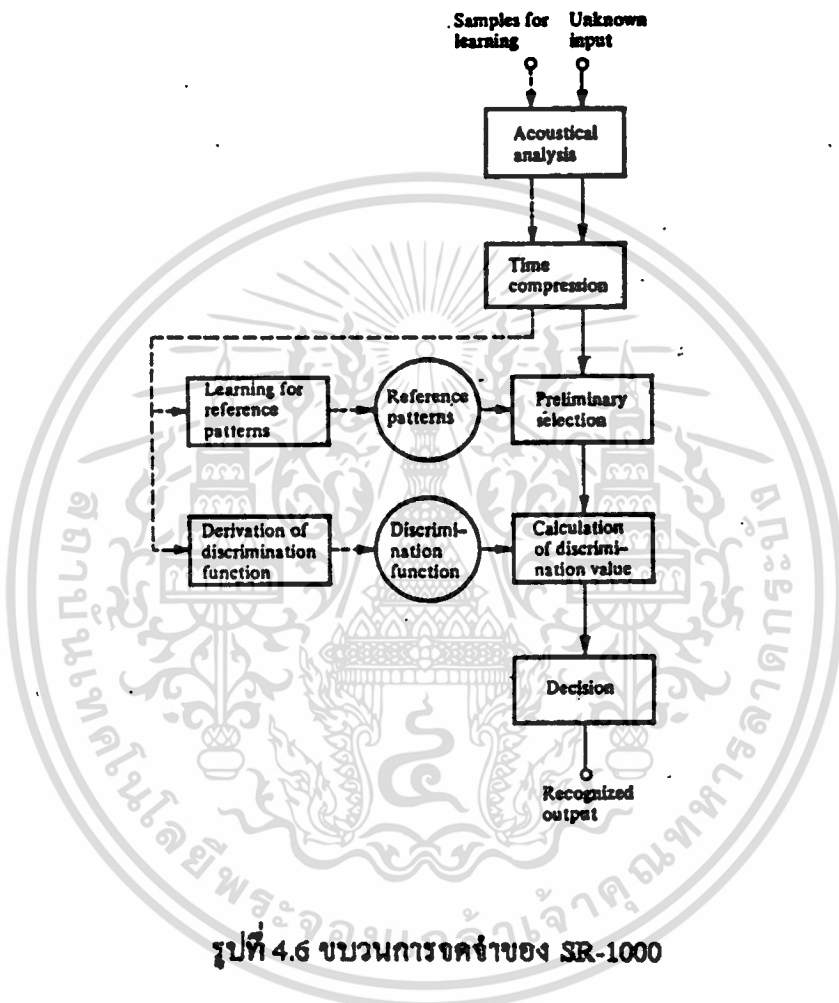
### 3. ผู้พูดไม่เฉพาะเจาะจง และประเภทรูปแบบการอ้างอิงเป็นคำ

ในระบบที่ใช้รูปแบบการอ้างอิงเป็นคำสำหรับผู้พูดไม่เฉพาะเจาะจง การวิเคราะห์ทางสถิติของข้อมูลต่างๆจากเสียงผู้พูดทั้งหลายจะมีในขั้นต่อไปแทนการเรียนรู้ในโหมด on-line การวิเคราะห์หมายถึง การหาฟังก์ชันความแตกต่างแบบ non-linear สำหรับแต่ละคำในศัพท์ที่ให้มา หรือการเตรียมหลายรูปแบบอ้างอิงสำหรับแต่ละคำ โดยจัดกลุ่มขึ้นกับข้อมูลจากเสียงพูดจากผู้พูดทั้งหลายต่อคำหนึ่ง โดยการจัดกลุ่ม

กระบวนการของรุ่น SR-1000 แสดงในรูปที่ 4.6 ในกระบวนการเรียนรู้จะได้ฟังก์ชัน discrimination แบบ non-linear และรูปแบบอ้างอิงสำหรับการเลือกในเบื้องต้นจากโหมด off-line ฟังก์ชัน discrimination ได้จากเซตของลิเนียร์ฟังก์ชัน โดยการใช้วิธี LPC รูปแบบอ้างอิงถูกใช้สำหรับการเลือกขั้นในเบื้องต้นและได้มาโดยการใช้เทคนิคการจัดกลุ่ม

ในระบบการจดจำทางปฏิบัติ การวิเคราะห์อรรถศิกและการบีบทางเวลาจะกระทำในเบื้องต้น การบีบทางเวลาเป็นกระบวนการลดผลกระทบค่าความแตกต่างของความเร็วในการพูดสำหรับผู้พูดแต่ละคน

ระบบของ Bell lab ใช้รูปแบบอ้างอิงเป็นจำนวน 12 รูปแบบต่อคำ ซึ่งรูปแบบได้จากการจัดกลุ่มชายหญิงผู้พูด 50 คน กระบวนการจดจำผ่านสายโทรศัพท์ที่ใช้การวิเคราะห์ LPC และการแมทช์แบบ DB :



รูปที่ 4.6 ขบวนการจดจำของ SR-1000

#### 4. ผู้พูดไม่เฉพาะเจาะจงและรูปแบบการอ้างอิงเป็นหน่วยเสียง

ตัวอย่างของประเภทนี้เป็นระบบการจดจำเสียงพูดของมหาวิทยาลัย Tohoku ในญี่ปุ่น หลังการวิเคราะห์ห่อคุณศัพท์ จุดยอดที่เป็น local ของสเปกตรัมซึ่งสัมพันธ์ใกล้ชิดกับความถี่และโมเมนต์ของสเปกตรัมจะถูกดึงออกมา การจดจำหน่วยเสียงกระทำโดยลักษณะนี้มีการเพิ่มเติมรูปแบบอ้างอิงสำหรับกลุ่มหน่วยเสียงที่พิเศษซึ่งมีบทบาทธรรมดา เช่น การหยุด และใช้หลายรูปแบบอ้างอิงสำหรับสระหนึ่งคำ ทั้งนี้เพราะสระมีผลต่อคุณลักษณะทั้งหลายของผู้พูดแต่ละคน อัตราการจดจำ 95.1 เปอร์เซ็นต์สำหรับ 51 คำ และ 89.0 เปอร์เซ็นต์สำหรับ 166 คำ

หลายวิธีการจดจำเสียงพูดที่ทดลอง มีการเพิ่มเติม สามารถสรุปดังต่อไปนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1. อัตราการจดจำทางปฏิบัติที่เหมาะสมได้มาจากการจดจำคำเดี่ยวๆที่มีรูปแบบอ้างอิงเป็นคำหรือหน่วยเสียง
2. อัตราการจดจำที่เป็นประโยชน์ในทางปฏิบัติสำหรับผู้พูดเฉพาะในจำนวนคำหลักร้อย
3. สำหรับผู้พูดไม่เจาะจง คำศัพท์ 10-50 คำ เป็นช่วงของอัตราการจดจำทางปฏิบัติ
4. การจดจำที่ต่อเนื่อง 10 ศัพท์และคำที่ใช้ควบคุมอื่นๆจำนวนน้อย เป็นขนาดของศัพท์ที่เป็นประโยชน์สำหรับอัตราการจดจำทางปฏิบัติ

#### 4.4 การสังเคราะห์เสียง

การจำแนกระดับของอุปกรณ์ตอบสนองทางโสตโดยใช้เทคนิคการวิเคราะห์ข้อมูลของเสียงตามตารางที่ 4.1 แบ่งได้เป็น 4 แบบ

Material	Method	
	Waveform coding	Source coding
Word, phrase	I. Editing prerecorded units	II. Analysis-synthesis
phonemes, syllables	III. Editing of speech element	IV. Synthesis by rule

ตารางที่ 4.1 การจำแนกอุปกรณ์ตอบสนองทางโสต

แบบที่ 1 จะสร้างเสียงพูดโดยการเชื่อมโยง (Concatenation) ของคำพูดหรือวลีที่เก็บไว้ในรูปแบบของ analog waveform หรือแบบ digit form ที่เป็นการเข้ารหัสแบบ PCM แบบนี้เรียกว่า Editing of prerecorded units เครื่องมือนี้ใช้ในระยะเวลาประกาศและคอนตักโทรศัพท์ภายหลังการสอบถาม การเข้ารหัส direct PCM จะต้องการข้อมูล 56-64 กิโลบิตต่อวินาที และมีเทคนิคการลดข้อมูลหลายแบบที่ใช้เช่น ADPCM, APC, SBC และ ATC ซึ่งจะช่วยลดอัตราส่งของข้อมูลเป็น 2.4-12 กิโลบิตต่อวินาที

แบบที่ 2 เรียกว่าแบบ analysis - synthesis หรือ editing of parameters แบบนี้มีพื้นฐานมาจากการเข้ารหัสแบบ linear predictive หรือเทคนิค spectrum processing แบบอื่นๆ ข้อมูลที่ถูกเก็บไว้จะควบคุมสัญญาณได้จากเครื่อง จำนวนข้อมูลที่ต้องใช้คือวินาทีจะน้อยกว่าแบบที่ 1 คือ 1.2-9.6 กิโลบิตต่อวินาที แต่คุณภาพเสียงที่ได้ดีกว่าแบบที่ 1 เล็กน้อย

แบบที่ 3 จะเป็นการเชื่อมเสียงของพยางค์เดี่ยว ซึ่งจะได้คุณภาพเสียงที่ไม่ดีนัก เพราะขาดการ coarticulation ระหว่างคุณลักษณะของพยางค์และ prosodic ของคำและประโยค ด้วยเหตุนี้มันจึงไม่ถูกใช้ในการปฏิบัติทางประยุกต์ อีกวิธีหนึ่ง ถูกทดลองโดยใช้หน่วยของเสียงพูดที่เล็กกว่าพยางค์หรือหน่วยเสียง หน่วยเสียงนี้จะถูกเก็บและเสียงที่ได้ถูกสร้างโดยการ concatenation ของแต่ละหน่วย ระบบนี้เรียกว่า editing of elementary units of speech ในการสร้างขบวนการของวิธีนี้ใช้หลายวิธีเช่น zero cross, impulse response ของ vocal fracts และองค์ประกอบของความถี่ที่เด่นในคลื่นเสียงพูด มีหลายกรณีที่ได้คุณภาพของเสียงที่ดีแค่ต้องใช้เสียงที่มีขนาดเล็กจำนวน 100-1000 หน่วย สำหรับการสังเคราะห์เสียงในประโยค การรวบรวมข้อมูลของลักษณะทางโพรไซคิก เป็นสิ่งจำเป็นรวมทั้งต้องการจำนวนของหน่วยของเสียงมากขึ้นด้วยในการใช้ระบบนี้จะไม่มีขีดจำกัดของเสียงพูดที่จะถูกสร้างด้วย เพราะโครงสร้างของฮาร์ดแวร์ที่เป็นแบบง่ายแต่คุณภาพจะดีน้อยกว่าการสังเคราะห์ในแบบอื่นๆ ดังนั้นวิธีนี้จึงไม่ถูกใช้ในทางปฏิบัติการประยุกต์ ในบางกรณีการรวมกันของแบบที่ 1 และแบบที่ 3 จะถูกใช้เป็นพื้นฐานให้แก่แบบที่ 3 เพื่อปรับปรุงคุณภาพของเสียง

แบบที่ 4 จะวิเคราะห์ตัวแปรของหน่วยของพยางค์หรือ phoneme เพื่อใช้ในการควบคุมข้อมูลของการสังเคราะห์เสียงพูด อุปกรณ์ตัวนี้จะสร้างค่าจากคำศัพท์ที่มีอยู่อย่างไม่จำกัดแต่ได้คุณภาพที่ไม่ดีเพราะว่าจะขาดคุณสมบัติ coarticulation และ prosodic เช่นเดียวกับอุปกรณ์ในแบบที่ 3 อย่างไรก็ตามแบบที่ 4 ใช้รูปแบบการสร้างเสียงพูดที่เป็น spectrum envelope และคุณลักษณะของที่มาของเสียงที่สามารถควบคุมได้อย่างอิสระและสามารถปรับปรุงคุณภาพของเสียงโดยคุณสมบัติตามกฎต่างๆที่สามารถได้คุณภาพเสียงที่มีความใกล้เคียงทางรูปแบบ coarticulation และ prosodic ของเสียงธรรมชาติ กรรมวิธีนี้เรียกว่า speech synthesis by rules ที่ซึ่งชุดของสัญลักษณ์สำหรับหน่วยเสียง พยางค์ หรืออักขระจะถูกปรับเข้าไปในฐานะของอินพุทและคลื่นของเสียงที่มีลักษณะตรงกันจะถูกสร้างขึ้นมา

กฎเกณฑ์สำหรับการจำลองเสียงสะท้อนโนเสียงพูดเกิดขึ้นจากสภาพแวดล้อม phonemic และ linguistic ในเสียงธรรมชาติเป็นส่วนสำคัญอย่างแท้จริงในการสังเคราะห์เสียงถึงแม้ว่าการสังเคราะห์โดยกฎเกณฑ์นี้จะเป็เป้าหมายสุดท้ายของการสังเคราะห์แต่เสียงพูดก็ถูกสร้างขึ้นมาโดยมีข้อจำกัดในคุณภาพ เนื่องจากพื้นฐานของกฎเกณฑ์นั่นเอง

Synthesis method	Type	Unit	Vocabulary	Hardware	Quality	Memory
Waveform coding	I.Editing prerecorded units	Words, phrases	Small	Simple	Good	Large
	II.Editing element	Syllables, phonemes	Large	Simple	Poor	Medium
Source coding	III. Analysis-synthesis	Words, Phases	Medium	Complex	Good	Medium
	IV. Synthesis by rules	Phonemes, syllables	Large	Complex	Fair	Small

ตารางที่ 4.2 คุณลักษณะของเครื่องมือตอบสนองต่อเสียง

คุณลักษณะของเครื่องมือตอบสนองเสียงต่างๆ ที่กล่าวมาแล้วจะแสดงดังในตารางที่ 4.2 จากตารางแต่ละระบบจะมีข้อได้เปรียบเสียเปรียบและทางประยุกต์ใช้งานเป็นของตัวเอง ไม่สามารถกล่าวได้ว่าแบบใดดีที่สุดเพราะว่ามันถูกออกแบบเพื่อให้มีความเหมาะสมในการใช้งานในแต่ละรูปแบบ โดยพิจารณาจากคุณลักษณะและข้อจำกัด

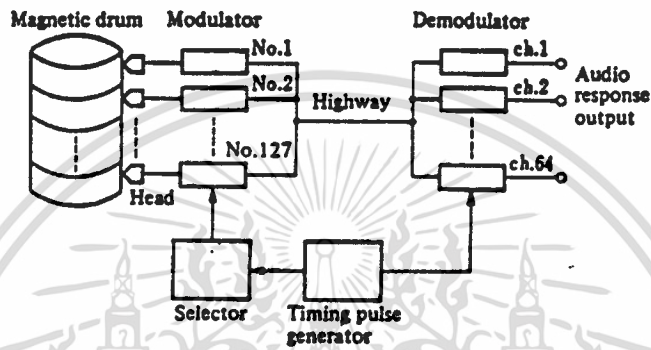
ระบบการตอบสนองทางเสียงทั้งหมดเคยมีความสัมพันธ์กันอย่างมากในด้านของ ฮาร์ดแวร์ และสเกล พวกมันต้องทำงานในโหมดมัลติเพล็กซ์ เพื่อที่จะได้รับผลการปฏิบัติงานอย่างคุ้มค่าและทางการ ใช้งานเป็นตัวทำให้เกิดข้อจำกัดทางการอำนวยความสะดวกในการให้บริการข้อมูลของเครือข่ายโทรศัพท์หรือเป็นเครื่องแสดงผลัพท์ของศูนย์คอมพิวเตอร์ขนาดใหญ่ อย่างไรก็ตาม เป็นที่น่าสังเกตว่าการพัฒนาในเทคนิค LSI วงจร gate จำนวนหลายๆเส้น สามารถบรรจุอยู่ได้ภายใน LSI เพียงชิปเดียวและ LSI สำหรับการวิเคราะห์สังเคราะห์และแม้แต่สังเคราะห์ด้วยกฎสามารถนำมาใช้ในทางปฏิบัติได้ ในทางประยุกต์ของการสังเคราะห์เสียงพูดจะมีการขยายไปได้กว้างขวางเพราะการสังเคราะห์หนึ่งช่องสัญญาณมีราคาถูก

## การสังเคราะห์เสียงมี 2 ประเภทใหญ่ๆด้วยกัน

1. การสังเคราะห์เสียงโดยวิธี Waveform Coding
2. การสังเคราะห์เสียงโดยวิธี Source Coding

### 1.การสังเคราะห์เสียงโดยวิธี Waveform Coding

#### 1.1 Speech Synthesis by The Editing of Prerecorded Words



รูปที่ 4.7 แสดงบล็อกโคอะแกรมของอุปกรณ์สังเคราะห์เสียง

รูปที่ 4.7 เป็นบล็อกโคอะแกรม ของอุปกรณ์สังเคราะห์เสียง ใช้ในการบริการสอบถามข้อมูลทางโทรศัพท์ คำและวลีจะถูกบันทึกใน magnetic drum (แสดงทางด้านซ้ายของรูป) และถูกสร้างใหม่โดย reproducing head บางตัวจะถูกใช้โดย modulated gate ส่งไปยัง transmission highway การเชื่อมต่อที่เกิดขึ้นที่สถานีรับและได้ผลลัพธ์ออกมาเป็นประโยค

#### 1.2 Speech Synthesis by The Editing of Speech Elements

โดยปกติหนึ่งช่วงระดับลูกคลื่นเสียงของคลื่นเสียงทุกถูกใช้เป็นหน่วยในการสังเคราะห์เสียงโดยอุปกรณ์บันทึกเสียง(editing of speech element) ขบวนการ zero-cross ถูกอธิบายในที่นี้เพราะเป็นกรณีที่ง่ายที่สุดในระบบนี้ หนึ่งช่วงระดับลูกคลื่นเสียงถูกเก็บเป็น speech element ในรูปของ zero-cross ในระหว่างอนุกรมแต่ละหน่วยเสียง  $i$  ให้แทนด้วย  $P_i$  จำนวนครั้งที่ทำซ้ำแทนด้วย  $r$  และแอมพลิจูด เป็น  $a$  จะเขียนได้ในรูปของ  $aP_i^r$  ตัวอย่างเช่น สระเสียง  $|a| = 10, r=8$  ดังนั้น  $|a| = 10P_i^8$  เสียงที่ไม่ใช่เสียงพูด อนุกรมของ zero-cross ในหน่วยความยาวจะใช้เป็นการสุ่มหรือหยุดในช่วงเวลาหนึ่งโดยไม่มีผลลัพธ์เติมในส่วนนั้น

ยิ่งไปกว่านั้นในการแสดงถึงแต่ละหน่วยเสียง การแสดงส่วนการเคลื่อนย้ายของ element จะมีความจำเป็นมาก การเคลื่อนย้ายนี้จะถูกสร้างจากการรวมกันของแบบที่ 1 และ แบบเอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

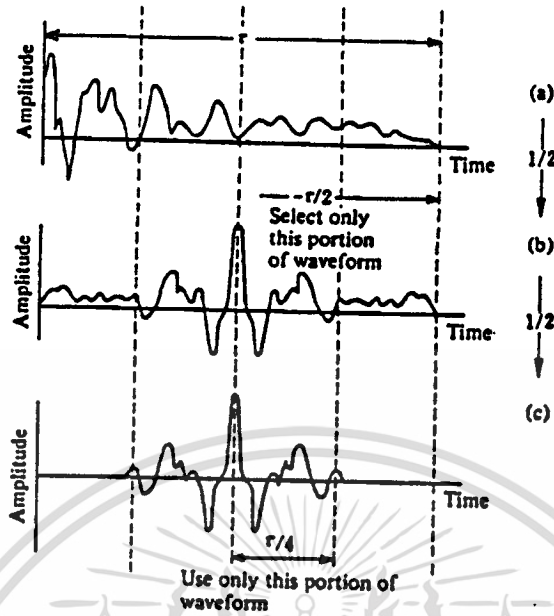
ที่ สอง 14 ความถี่ในช่วง 300-950 เฮิรท์ และ 17 ความถี่ระหว่าง 700-2500 เฮิรท์ จะถูกกำหนดในแต่ละแบบ

ในระบบระดับเสียงถูกควบคุมเพียงสองระดับ คือระดับสูงและระดับต่ำ ระดับสูงจะถูกประมาณการโดยการคัดส่วนท้ายของ element และระดับเสียงต่ำจะถูกประมาณโดยใช้ element เดิมส่วน

แม้แต่ในส่วนของการควบคุมแอมพลิจูดด้วยเวลาคงที่ที่แน่นอนในการสร้างเสียงพูดขึ้นมาใหม่ ผลลัพธ์จะต้องการการปรับปรุงในคุณภาพ เพราะการสังเคราะห์ข้อมูลส่วนใหญ่จะเป็นเพียงการใช้ช่วงเวลาใน zero cross และเป็นการควบคุมโพไซติกแบบง่าย ๆ เท่านั้น

การปรับปรุงในวิธีการ speech element เทคนิคต่างๆ ถูกนำมาประยุกต์ใช้ ช่วงเวลาและข้อมูลแอมพลิจูดจะถูกดึงออกมาจากเสียงพูดตามธรรมชาติ ที่ซึ่ง element ของหนึ่งระดับเสียงจะถูกใช้สำหรับการสังเคราะห์เสียง วิธีการนี้พยายามที่จะผสมผสานกันระหว่างการแก้ไขเสียงที่ถูกบันทึกไว้กับการสังเคราะห์ element เสียง การผสมผสานกันนี้ถึงแม้ว่าคำศัพท์จะมีข้อจำกัดแต่ความจำเป็นในการสังเคราะห์จะลดลง

ในระบบนี้จะมี 3 กระบวนการที่ใช้ในการลดอัตราข้อมูลของหน่วยความจำสำหรับเสียงพูด อย่างแรกคือการทำซ้ำของหนึ่งระดับเสียง 2-3 ครั้งในการสั้นของเสียงที่คงที่และส่วนที่คล่องของมัน ตัวเลขที่ยอมรับได้ในการเกิดซ้ำกันคือประมาณ 3-4 สำหรับเสียงพูด และ 7-8 สำหรับเสียงที่ไม่ใช่เสียงพูด อย่างที่สองเป็นกระบวนการของรูปคลื่นระดับเสียงในช่วงเวลาหนึ่งๆ การ pitch period waveform สามารถทำให้อยู่ในขอบรูปคลื่นที่สมดุลในจุดกึ่งกลางของเวลา ดังแสดงในรูปที่ 4.8 อย่างที่สามเป็นการลดแอมพลิจูดรูปคลื่นให้เป็นศูนย์ทั้งในส่วนเริ่มต้นและสิ้นสุดของหนึ่งในสี่ส่วนของหนึ่งระดับเสียงเต็ม



รูปที่ 4.8 แสดงรูปคลื่นของระดับเสียง

## 2. การสังเคราะห์สัญญาณเสียงโดยวิธี Source coding

สัญญาณเสียงจะเป็นเสียงที่ออกมาจากริมฝีปาก คั้นกำเนิดของมันคือการสั่นสะเทือนของสัญญาณเสียงที่ต่อเนื่องกันใกล้กับจุดหดรัดตัวเพื่อการออกเสียง และจะถูกถ่ายทอดมาทางหลอดลมหรือหลอดเสียง การออกเสียงจะถูกนำไปเปลี่ยนแปลงโดยรูปร่างของหลอดเสียงหรือหลอดลมและถ่ายทอดเสียงออกไป ซึ่งหลอดลมนั้นจะให้คุณสมบัติของความถี่หลายๆความถี่ในแต่ละเสียง ซึ่งจากลักษณะนี้คือการออกเสียงทางกลศาสตร์ และการสังเคราะห์สัญญาณเสียงจะขึ้นกับลักษณะดังกล่าวมาแล้วนี้ เรียกว่าการสังเคราะห์โดยการ Source coding มีวิธีการ Source coding หลายชนิดแต่ที่นิยมมี 3 ชนิดคือ

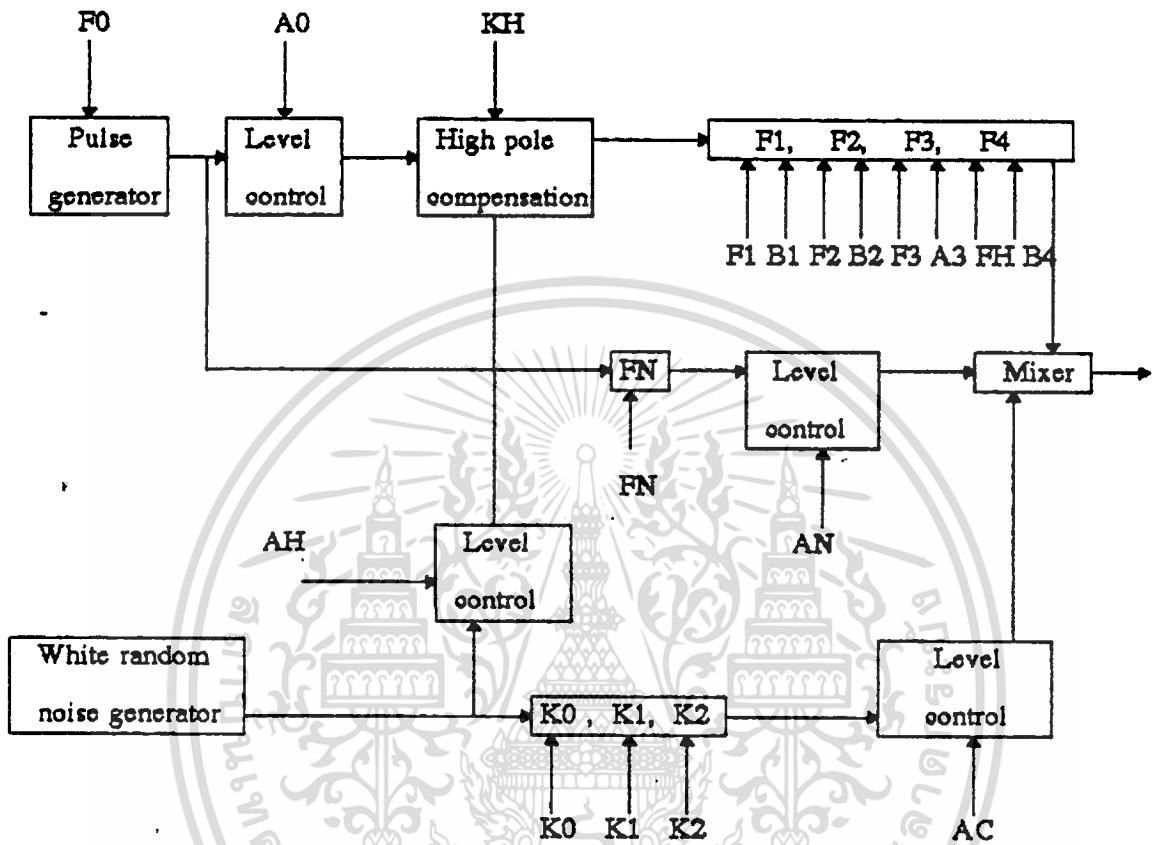
### 2.1.แบบ Formant

จะเป็นแบบที่จำลองการทำงานแบบ Transfer function ของหลอดเสียงโดยใช้วิธีการคั่นกันของวงจรเสียงก้อง เสียงไม่ก้องหลายๆวงจรแบบอนุกรมและแบบขนาน ซึ่งความถี่และค่าแบนด์วิธจะถูกควบคุมแบบอิลคทรอนิกส์

แบบOVE - III ของ Royal of Teohnology ของสวีเดนเป็นตัวอย่างของการต่อแบบอนุกรม แต่แบบของ Joint Speecch Research Unit (JSRU) ของอังกฤษจะเป็นตัวอย่างของแบบขนาน แบบแรกนั้นจะค้ำสำหรับการสังเคราะห์เสียง แต่แบบหลังจะค้ำสำหรับการวิเคราะห์ที่ไม่ใช่เสียง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แบบอนุกรมของการสังเคราะห์แสดงดังรูปที่ 4.9 และแสดงรูปแบบการควบคุม สำหรับชนิดนี้ดังตารางที่ 4.1 ซึ่งมีทั้งหมด 17 พารามิเตอร์



รูปที่ 4.9 แสดงการสังเคราะห์เสียงแบบอนุกรม

การสังเคราะห์เสียงแบบ Formant นี้สามารถผลิตเสียงที่มีคุณภาพดีขนาด 2000 บิตต่อวินาที แต่มันจะใช้เวลามากและกระบวนการที่ซับซ้อนกว่าเพื่อที่จะทำการควบคุมข้อมูล จากการวิเคราะห์สัญญาณรูปคลื่นเสียงจริง ขบวนการสัญญาณเสียงที่ถูกใช้ในอดีต โดยวิธีการ อ่านจากรูปแบบสัญญาณเสียง หรือการวิเคราะห์โดยสังเคราะห์ วิธีการนี้จะถูกนำมาใช้กันอย่าง แพร่หลาย และถูกนำมาวิจัยเสียงเป็นเวลาหลายปี เมื่อไม่นานมานี้ VOTRAX ของบริษัท Vocal interface กับบริษัท Intel ได้วิธีนี้ในการวิเคราะห์เสียง

Name	Number of bits	Range	Step of control	Remarks
FO	0-5	50-300 Hz	3%	Pitch frequency
F1	0-5	200-1200 Hz	3%	Formants for vowel
F2	0-5	500-3100 Hz	3%	
F3	0-5	1000-6200 Hz	3%	
A0	0-5	32 dB	0.5 dB	
AC	3-5	28 dB	4 dB	Level of fricatives
AH	0-2	28 dB	4 dB	Level of aspiration
AN	4-5	24 dB	8 dB	Level of nasal sound
FN	0-3	200-1100 Hz	12%	Nasal formant.
FH	2-5	2500-6200 Hz	12%	F4 and higher poles
K0	0-5	1000-6200 Hz	3%	Zero of fricative
K1	0-5	1000-6200 Hz	3%	Formants of consonant
K2	0-5	2500-9800 Hz	3%	
B1	1-2		100 Hz	Bandwidth of vowel formants
B2	4-5		100 Hz	
B3	0		200 Hz	
B4	3		200 Hz	

ตารางที่ 4.3 แสดงรูปแบบการควบคุมการสังเคราะห์แบบอนุกรม

## 2.2. แบบ PARCOR

เมื่อมีขนาดสัญญาณแอมป์ถึง 6.4-10 กิโลเฮิร์ต ถูกนำมาใช้เพื่อให้ได้การวิเคราะห์เสียงที่มีคุณภาพ มันจะถูกเลือกตามจุดประสงค์ของการวิเคราะห์ เพื่อให้ได้การวิเคราะห์เสียงตามคุณภาพ ค่าตั้งที่ใช้จะมีผลต่อประสิทธิภาพของ PARCOR จากตารางที่ 4.4 จะเป็นการแสดงให้เห็นถึงคุณสมบัติต่างๆ เพื่อให้เสียงมีคุณภาพที่ดีขึ้นเมื่อนำไปวิเคราะห์

Transmission rate (bits/sec)	Frame period (msec)	Order of analysis	Bit for $k_i$ (bit)										Bit for source (bits)		
			1	2	3	4	5	6	7	8	9	10	Total	Pitch frequency	Source amplitude
9600	10	10	12	9	8	8	8	8	8	7	7	7	82	7	7
4800	15	10	9	7	6	6	5	5	5	5	5	5	58	7	7
2400	20	10	7	5	3	3	3	3	3	3	3	3	36	6	6

ตารางที่ 4.4 แสดงคุณสมบัติต่างๆ เพื่อให้เสียงมีคุณภาพดีขึ้น

## 2.3 แบบ Vocal Tract Analog

จะใช้ลักษณะของการจำลองแบบของการส่งผ่านหลอดเสียงโดยใช้ฟังก์ชันของพื้นที่ภาคบาท โดยมากเสียงของผู้ชายมักมีขนาดราว 17 เซนติเมตร จากริมฝีปากถึงช่องว่างระหว่างเอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เส้นเสียง และส่วนของ 17 เซนติเมตร นี้จะทำให้สามารถประมาณได้ว่าขนาดพื้นที่กากบาทของหลอดเสียงจะมีขนาดเส้นผ่านศูนย์กลางราว 1 เซนติเมตร ซึ่งถูกนำมาต่อกันแบบอนุกรม 4 ตัว ด้วยวงจร L และ C ในส่วนของริมฝีปากซึ่งเป็นด้านปลายของท่อจะถูกต่อโดยวงจรมวลของ L และ R โดยตัวกำเนิดเสียงจะถูกส่งไปยังตัวอินพุทของการต่ออนุกรมวงจร  $\pi$  type และสัญญาณเสียงของ unvoice sound จะถูกนำมาต่อกับวงจรเสียงเพื่อต้นสะเทือน ปัญหาของเสียงที่วิเคราะห์มีคุณภาพต่ำเนื่องมาจาก

1. ความถูกต้องแม่นยำของหลอดเสียงค่อนข้างต่ำรวมถึงอวัยวะการออกเสียงที่เคลื่อนที่อื่นๆ

2. ขาดความถูกต้องแม่นยำ ของข้อมูลบนฟังก์ชันพื้นที่กากบาท

3. ยากต่อการสกัดความเปลี่ยนแปลงชั่วคราวของ ฟังก์ชันพื้นที่กากบาท

เทคนิคของการตรวจสอบความเคลื่อนไหวของหลอดเสียงและการวิเคราะห์แบบอัตโนมัติจะถูกนำมาพัฒนาต่อไปในอนาคตโดยจะเป็นการรวมกันของรูปแบบของหลอดเสียงกับ 2 ส่วนสำคัญของการเชื่อมโยงการต้นสะเทือน

#### 4.4 ระบบการคิดต่อสื่อสาร

โดยทั่วไปเราแบ่งการทำงานออกเป็น 3 ส่วนหลัก(โหมดของการคิดต่อสื่อสาร)ในการคิดต่อระหว่างมนุษย์กับเครื่องจักร โดยเสียงคือ

1. ระบบการตอบสนองเสียง(Voice Response System)
2. ระบบการจดจำผู้พูด(Speaker Recognition System)
3. ระบบการจดจำเสียง(Speech Recognition System)

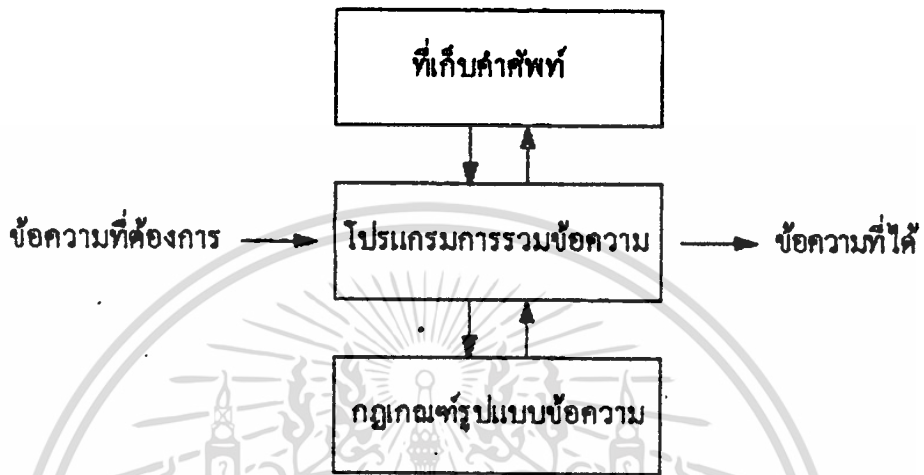
ในส่วนแรกคือ ระบบการตอบสนองเสียงนั้น จะเป็นส่วนที่ใช้ในการคิดต่อทางเดียวคือ จากเครื่องจักรไปยังมนุษย์ โดยจะเป็นการตอบสนองการทำงานที่รับมาจากสัญญาณเสียงพูด

ในส่วนที่สองและสามนั้น จะเป็นส่วนของการคิดต่อโดยเสียงของมนุษย์ไปยังเครื่องจักร โดยในส่วนที่สองระบบการจดจำผู้พูดนั้น จะมีหน้าที่ในด้านของการพิสูจน์เพื่อระบุตัวผู้พูด (เช่น การตัดสินใจว่า ใช่หรือไม่ใช่ตัวผู้พูดที่อ้างอิงถึง) หรือการพิสูจน์ตัวผู้พูดจากข้อมูลที่เก็บไว้ ดังนั้นระบบการจดจำผู้พูดจึงแบ่งเป็นส่วนของการพิสูจน์ผู้พูดกับส่วนของการระบุตัวผู้พูดต่อไป

ในส่วนที่สามคือส่วนของการจดจำเสียงพูดนั้น สามารถแบ่งเป็นส่วนย่อยๆที่ขึ้นกับแฟคเตอร์เช่น ขนาดของคำศัพท์ ตัวผู้พูด สภาพการณ์ของการพูด เป็นต้น หน้าที่พื้นฐานของการจดจำเสียงพูดนั้นคือ จดจำลักษณะของการออกเสียงทั้งหมดอย่างแม่นยำ เช่น สำเนียงพูดหรือการสะกดอักษรของการพูด ไปยังรูปแบบของระบบการพิมพ์ หรือการเข้าใจคำพูดของผู้พูด เช่นมี

การตอบสนองอย่างถูกต้องสิ่งที่พูด ในส่วนของการเข้าใจคำพูดนั้นมีความสำคัญที่สุดต่อระบบที่ใช้กับระบบที่มีขนาดค่าใหญ่และต่อเนื่องกัน แต่ในส่วนของการจดจำจะสำคัญที่สุดต่อระบบคำศัพท์ที่จำกัดจำนวนผู้พูดน้อยและจำนวนคำแตกต่างกันน้อย

### 1. ระบบการตอบสนองเสียง



รูปที่ 4.10 แสดงบล็อก โดอะแกรมของระบบการตอบสนองเสียง

จากรูปที่ 4.10 แสดงบล็อก โดอะแกรมของระบบการตอบสนองเสียงของคอมพิวเตอร์ทั่วไปซึ่งมีลักษณะดังนี้

1. ส่วนจัดเตรียมสำหรับเก็บข้อมูลของคำศัพท์สำหรับระบบการตอบสนองเสียง
2. กฎเกณฑ์สำหรับกำหนดรูปแบบของข้อความจากรากศัพท์ของคำศัพท์
3. โปรแกรมสำหรับการรวมข้อความตอบสนองเสียง

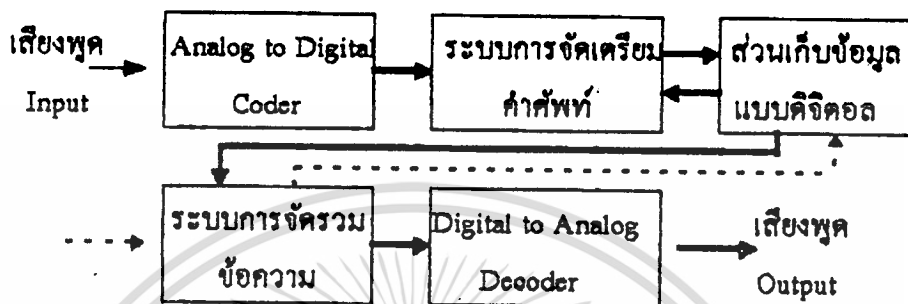
มีการเข้าถึงการดำเนินการของระบบการตอบสนองเสียง 2 อย่างหลักๆคือ

1. การพยายามในการสร้างเครื่องจักรที่มีความสามารถในการเปรียบเทียบเสียงพูดได้เหมือนมนุษย์ ซึ่งสามารถที่จะผลิตข้อความตามรูปแบบโดยมีสัญญาณควบคุมตามต้องการ
  2. ระบบคำศัพท์ที่จำกัดข้อความที่ได้ถูกสร้างโดยการรวมคำศัพท์ที่แยกเป็นคำๆจากที่เก็บคำศัพท์
- รูปที่ 4.11 แสดงถึงบล็อก โดอะแกรมของระบบการตอบสนองเสียงแบบคิจิตอล โดยคำศัพท์ที่ประกอบด้วยคำแยกโดดกับกลุ่มคำที่แสดงในรูปของคิจิตอลและเก็บอยู่ในหน่วยความจำแบบคิจิตอล โดยข้อความจะถูกผลิตโดยการนำคำและกลุ่มคำจากที่เก็บมาผลิตโดยลำดับความเหมาะสมและมีสิ่งที่จะต้องพิจารณาอยู่ 3 อย่าง ในการออกแบบระบบตอบสนองเสียง คือ

- 1 การแสดงและการเก็บคำศัพท์พื้นฐานต้องถูกเลือกและระบบที่ใช้ในการเข้าถึงแต่ละคำศัพท์ต้องง่าย

2 ความสำคัญของการตัดต่อแก้ไขการบันทึกเสียงเพื่อเลือกคำศัพท์จะถูกเตรียม

3 ต้องการระบบสำหรับการเลือกและผลิตคำศัพท์ตามลำดับที่กำหนด



รูปที่ 4.11 แสดงบล็อกโคโอะแกรมของระบบการคอบสนองเสียงแบบดิจิทัล

- → DIGITAL SPEECH REPRESENTATION
- CONTROL SIGNALS
- ANALOG SPEECH SIGNALS

	แบบ waveform coding	แบบ analysis/synthesis
-ขนาด bit	ต้องใช้ขนาดจำนวนบิตต่อหน่วยเวลามากกว่า (ขนาด 100 kb/s ถึง 1 Mb/s)	น้อยกว่า (ขนาด 600b/s ถึง 10 kb/s)
-ความซับซ้อน	น้อยถึงปานกลาง	ค่อนข้างมากถึงมาก
-ความยืดหยุ่นในการใช้งาน	น้อย	มาก

ตารางที่ 4.5 แสดงการเปรียบเทียบระหว่างการเปลี่ยนแปลงแบบ waveform coding กับ analysis/synthesis

ในการเลือกใช้วิธีการแปลงแบบดิจิทัลนั้นให้ดูจาก 3 กรณีเพื่อช่วยในการตัดสินใจ คือ

1. ขนาดบิตที่ต้องการอย่างน้อยที่สุดที่สามารถยอมรับได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2. ความซับซ้อนจากการแปลงไปและแปลงกลับสัญญาณ

3. ความยืดหยุ่นของการแสดงผล โดยความเป็นไปได้สำหรับการออกแบบตามราก

ศัพท์

จากตารางที่ 4.5 เป็นการเปรียบเทียบระหว่างการแปลงแบบ waveform กับแบบ analysis/synthesis แบบ waveform นั้นจะต้องการอัตราบิตมาก ต้องการพื้นที่ในการเก็บข้อมูลมากและมีอัลกอริทึมของการแปลงไปและแปลงกลับที่ง่ายขึ้น ไม่ซับซ้อนแค่ในแบบ analysis/synthesis จะใช้วิธีการแยกสัญญาณเสียงออกเป็นส่วนๆ ซึ่งจะทำให้มีศักยภาพในการทำงานสูงสุดในการใช้คำศัพท์ ซึ่งจะนำไปใช้ประโยชน์หลายด้านได้ ในส่วนของการหาจุดเริ่มต้นและจุดสิ้นสุดของสัญญาณเสียงของแต่ละคำนั้น เราสามารถใช้อัลกอริทึมของ short time energy ได้ โดยเป็นการคำนวณค่าของพลังงานจากจุดเริ่มต้น หรือจุดสุดท้ายของขนาดที่ทำการสุ่มจับสัญญาณ แล้วนำค่าที่ได้มาเปรียบเทียบกับขณะที่มีสัญญาณก่อนที่จะมีสัญญาณเข้าว่ามีค่าระดับพลังงานมากพอที่จะนำไปทำงานได้ จากการทดลอง ค่าพลังงานของสัญญาณที่เข้ามาจะมีค่าเป็น 3 เท่าของสัญญาณปกติก่อนสัญญาณอื่นๆจะเข้ามา

ในส่วนของการวิเคราะห์เสียงแบบต่อเนื่องนั้นเราใช้วิธีการของ analysis/synthesis เพื่อแยกเสียงออกเป็นคำช่วง pitch มีเสียงหรือไม่มีเสียง ความเข้มเสียง ความถี่สัญญาณเสียง เพื่อแยกคำแต่ละคำของคำพูดต่อเนื่องออกจากกันจากรูปที่ 4.12 จะเป็นรูปของการเปรียบเทียบสถาปัตยกรรม รูปที่ 4.12a จะเป็นรูปสัญญาณของเสียงที่เข้ามา รูปที่ 4.12b จะเป็นรูปของการแยกเสียงออกเป็นคำตามจังหวะเวลาและขนาด pitch ส่วนในรูปที่ 4.12c จะเป็นการขยายสัญญาณเสียงโดยระยะ pitch เท่ากัน ระยะห่างระหว่างคำไม่เปลี่ยนแปลง ในส่วนของการนำระบบการตอบสนองเสียงนั้น เราสามารถนำไปประยุกต์ใช้ได้หลายอย่าง อาทิเช่น

1. อุปกรณ์การติดต่อสื่อสารทางสายสัญญาณ ซึ่งจะช่วยให้ใช้เวลาในการทำงานได้มากขึ้น โดยไม่ต้องพะวง เมื่อเกิดการผิดพลาดและจะให้ผลการทำงานที่เร็วกว่า

2. ระบบการให้ข้อมูลและการบริการของสายการบิน

3. ระบบบอกราคาของสินค้าคงคลัง

4. ระบบการช่วยเหลือในการให้ข้อมูล

5. ระบบการให้ข้อมูลการทดสอบ

## 2. ระบบการจดจำผู้พูด

จะใช้วิธีการพิสูจน์โดยใช้พีชคณิต แบบบูลีนว่า ใช่หรือไม่ใช่ในการตรวจสอบเช่น พิสูจน์ว่า คำผู้พูด  $i$  ถ้า  $p_i(x) > \theta_i$  และปฏิเสธคำผู้พูด  $i$  ถ้า  $p_i(x) < \theta_i$   $p_{\infty}(x)$

เมื่อ  $p_1(x)$  คือค่าความน่าจะเป็นในการกระจายสำหรับการวัดค่า เวกเตอร์  $x$  สำหรับ  
ผู้พูด I

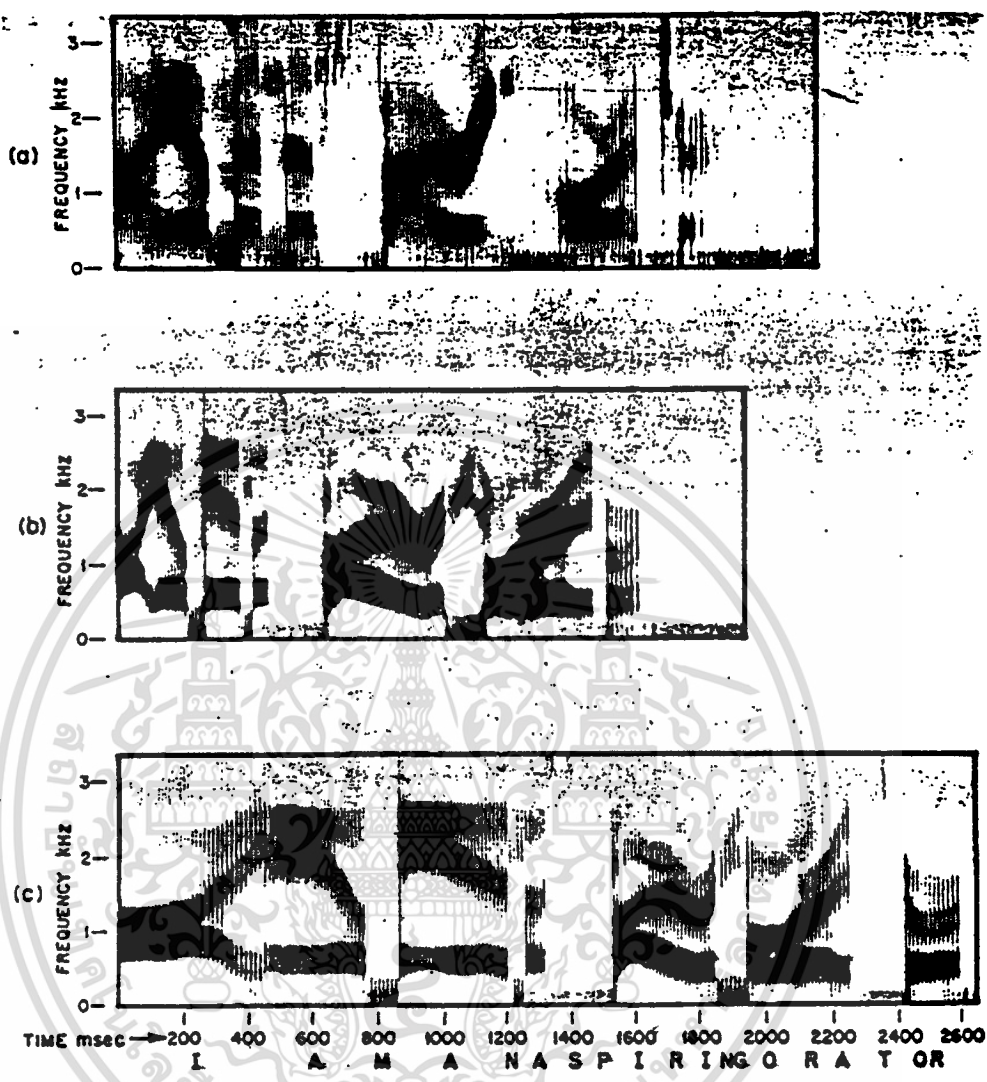
$q_1(x)$  คือค่าคงที่สำหรับผู้พูด I ซึ่งจะเป็นตัวควบคุมการผิดพลาดที่เกิดขึ้น

$p_{12}(x)$  คือค่าเฉลี่ยของความน่าจะเป็นในการกระจายสำหรับการวัดค่าเวกเตอร์  $x$   
สำหรับผู้พูดโดยทั่วๆไป

ซึ่งถ้าบังชี้ตัวผู้พูดนั้นจะทำให้้ง่ายในกรณีที่มีตัวผู้พูดน้อย แต่จะบังชี้ตัวผู้พูดได้ยาก  
ขึ้น เมื่อมีจำนวนผู้พูดที่เราเก็บไว้มากขึ้น แต่จะเป็นข้อดีในกรณีที่สามารภใช้ผู้พูดได้หลายคน ใน  
กรณีนำไปใช้งาน ในส่วนของการพิสูจน์ตัวผู้พูดจะกล่าวถึงอย่างย่อๆ เพราะโครงการนี้ไม่ได้ใช้  
กับผู้พูดหลายๆคน แต่ใช้กับผู้พูดคนเดียว

### 3. ระบบการจดจำเสียงพูด

ซึ่งจะแบ่งเป็น 2 กรณีคือ การจดจำคำแยกโคคๆ หรือกลุ่มคำกับกรณีการจดจำแบบ  
ต่อเนื่อง ในโครงการนี้จะใช้เฉพาะในกรณีแรก ซึ่งเราสามารถที่จะจดจำคำพูดได้โดยการบันทึก  
ข้อมูลจากเสียงพูดต่างๆลงในที่เก็บ เช่นการเก็บขนาดของคำ ค่าระดับของพลังงานเสียง ค่า  
ความถี่ของเสียงที่ได้จากการแปลงฟูเรียร์ ดังที่ได้กล่าวมาแล้ว การเก็บค่าความถี่ของเสียงแบบ  
ฮาร์โมนิกส์ ของสัญญาณเสียงถูกแรก การเก็บคำนี้จะนำค่าที่ได้มาหาค่าเฉลี่ยโดยการคำนวณค่า  
โดยค่าที่คำนวณมานั้นมาจากการหาค่าเฉลี่ยโดยตรง การหาค่าเฉลี่ยของค่าเบี่ยงเบนมาตรฐาน  
หรืออาจใช้วิธีอื่นๆทางสถิติก็ได้ เพื่อหาค่าความถูกต้องตามหลักของการทำงาน เพื่อให้การ  
ทำงานสามารถทำงานได้อย่างมีประสิทธิภาพสูงสุด



รูปที่ 4.12 แสดงรูปการเปรียบเทียบสเปกตรัมของเสียง

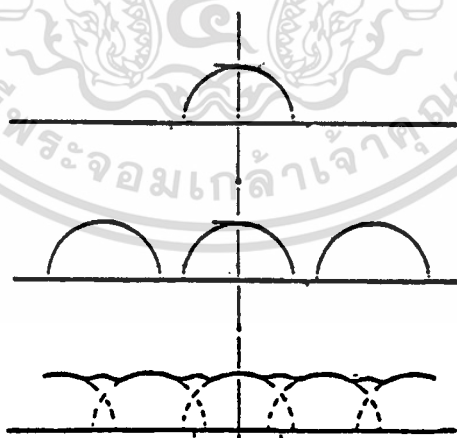
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
 ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## บทที่ 5

### แนวทางการออกแบบฮาร์ดแวร์

ฮาร์ดแวร์ที่ออกแบบจะนำมาใช้ในการเป็นส่วนเชื่อมต่อ(interface)เครื่องคอมพิวเตอร์กับเสียงพูด เพื่อให้สามารถนำข้อมูลของเสียงพูดเข้ามามีใช้ในการคำนวณ และการวิเคราะห์โดยคอมพิวเตอร์ได้ การบันทึกข้อมูลเสียงพูดเข้าสู่คอมพิวเตอร์จะต้องแปลงเป็นสัญญาณเสียงพูดไปเป็นสัญญาณดิจิทัลแล้วเก็บข้อมูลที่ได้ลงหน่วยความจำของฮาร์ดแวร์เมื่อหน่วยความจำบันทึกข้อมูลเต็มแล้วจะถ่ายข้อมูลทั้งหมดเข้าสู่คอมพิวเตอร์เพื่อนำไปใช้ในการคำนวณ

หลักการออกแบบฮาร์ดแวร์ ฮาร์ดแวร์จะต้องสามารถแปลงข้อมูลเสียงพูดโดยแปลงจากสัญญาณอนาล็อก (analog signal) ไปเป็นสัญญาณดิจิทัล (digital signal) ใช้แซมปลิงเรทค่า 10 กิโลเฮิร์ต ซึ่งค่าที่ใช้มาจากทฤษฎีการสุ่มตัวอย่างคือถ้าสัญญาณอนาล็อกที่ต้องการสุ่มตัวอย่างมีแถบความถี่ปฏิบัติงาน (bandwidth) เท่ากับ  $f_0$  แล้วจะต้องใช้ความถี่ในการสุ่มตัวอย่างมากกว่าหรือเท่ากับ 2 เท่าของความถี่  $f_0$  เพื่อที่จะได้สัญญาณดิจิทัลโดยไม่สูญเสียเนื้อหาสำคัญ หรือ ข้อมูลสำคัญรวมทั้งป้องกันการเกิดผลการเอเลตซิ่ง (aliasing effect) ซึ่งเป็นการซ้อนทับของสัญญาณดิจิทัลที่ได้



รูปที่ 5.1 แสดงการเกิด aliasing effect

เนื่องจากความถี่เสียงพูดของคนทั่วไปมีค่าสูงสุดประมาณ 4-5 กิโลเฮิร์ต ดังนั้นค่าความถี่ที่ใช้ในการสุ่มตัวอย่างควรจะมีความถี่ตั้งแต่ 8-10 กิโลเฮิร์ตขึ้นไป ซึ่งถ้าค่าความถี่ที่ใช้ในการสุ่มตัวอย่างมีค่าสูงๆ ก็จะได้รูปสัญญาณที่มีคุณภาพดีขึ้นตาม แต่ก็จะมีข้อเสียเวลามากขึ้นเช่นกัน เพราะจำนวนข้อมูลที่ได้จากการสุ่มตัวอย่างมีจำนวนมากขึ้นในขณะที่สัญญาณยังมีขนาดเท่าเดิม ดังนั้นเราจึงต้องพิจารณาเลือกความถี่ในการสุ่มตัวอย่างที่สามารถสุ่มตัวอย่างสัญญาณให้ได้รายละเอียดเพียงพอต่อการคำนวณแต่ไม่มากจนเกินไป เช่นการสุ่มตัวอย่างสัญญาณข้อมูล 10 ครั้งต่อหนึ่งคาบ อาจจะได้ข้อมูลที่ให้รายละเอียดเพียงพอต่อการคำนวณไม่จำเป็นต้องสุ่มตัวอย่างข้อมูลจำนวน 100 ครั้งต่อคาบ

การเลือกขนาดหน่วยความจำ จะต้องคำนึงว่าจะสามารถมีขนาดพอเพียงที่จะเก็บจำนวนข้อมูลที่ต้องการใช้ เช่นในการบันทึกข้อมูลเสียงพูด 1 คำ โดยที่ปกติการพูด 1 คำพูดนี้จะใช้เวลาไม่เกิน 1 วินาที การออกแบบขนาดหน่วยความจำจึงต้องพิจารณาให้สามารถบันทึกข้อมูลเสียงพูดได้ทั้งหมดคือมากกว่า 1 วินาทีขึ้นไปแต่ก็ต้องพิจารณาเนื้อที่ที่ใช้ในการเก็บข้อมูลด้วย ถ้าใช้หน่วยความจำมากจนเกินไปก็จะเปลืองเนื้อที่ในการเก็บข้อมูล

โครงการนี้วิเคราะห์ข้อมูลเป็นคำๆ ดังนั้นฮาร์ดแวร์นี้จึงเลือกใช้หน่วยความจำขนาด 20 กิโลไบต์ และใช้ความถี่ในการสุ่มตัวอย่างที่ 10 กิโลเฮิร์ต จะทำให้สามารถบันทึกข้อมูลเสียงพูดได้ 2 วินาที ซึ่งเพียงพอต่อการเก็บข้อมูลเนื่องจากเสียงคนพูดปกติจะใช้เวลา 1 วินาทีสำหรับคำหนึ่งคำ

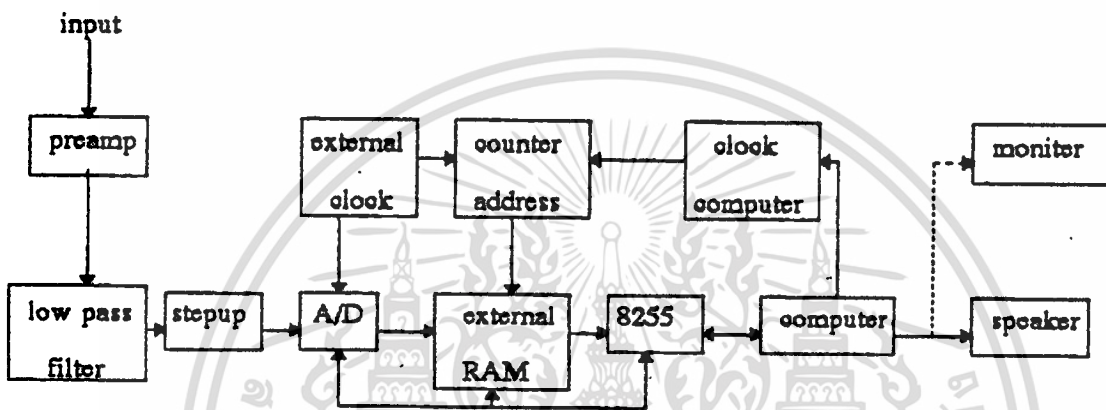
เนื่องจากอุปกรณ์ฮาร์ดแวร์ที่ออกแบบไว้สำหรับใช้งานต่อเชื่อมกับเครื่องคอมพิวเตอร์โดยทำเป็นการต่อที่ใช้ติดต่อกับพอร์ตต่อขยาย (expansion slot) ของคอมพิวเตอร์ ดังนั้นการออกแบบวงจรและอุปกรณ์จะต้องคำนึงถึงด้านต่างๆ คือ

1. จะต้องป้องกันไม่ให้ฮาร์ดแวร์นี้ ไปรบกวนการทำงานของระบบคอมพิวเตอร์ทั้งการป้องกันไม่ให้เครื่องคอมพิวเตอร์มารบกวนระบบของฮาร์ดแวร์ ซึ่งส่วนนี้จะใช้การใส่คิเค็ดแอกเครสของฮาร์ดแวร์ทางพอร์ตอินพุตเอาต์พุต (I/O port) เพื่อเอนเนเบิล (enable) หรือดิสเอนเนเบิล (disable) การติดต่อกับฮาร์ดแวร์เมื่อคอมพิวเตอร์ต้องการเรียกใช้งานก็จะอ้างถึงแอกเครสของฮาร์ดแวร์นี้จึงจะใช้งานได้เท่านั้น

2. จะต้องออกแบบวงจรอย่างรัดกุมและรอบคอบ เพราะความผิดพลาดของฮาร์ดแวร์จะทำให้มีผลกระทบต่อเครื่องคอมพิวเตอร์ถึงขั้นเสียหายได้ เช่น การลัดวงจรของฮาร์ดแวร์ การส่งสัญญาณที่ไม่ถูกต้องเข้าสู่คอมพิวเตอร์ เป็นต้น จึงควรป้องกันไว้โดยการใส่บัฟเฟอร์ให้กับเส้นสัญญาณต่างๆ ที่เชื่อมต่อกันระหว่างคอมพิวเตอร์กับฮาร์ดแวร์เมื่อเกิดการผิดพลาดขึ้นก็จะมีส่วนของบัฟเฟอร์กันเอาไว้ก่อน

3. เนื่องจากวงจรมีทั้งส่วนที่เป็นวงจรทางอนาล็อก และส่วนที่เป็นวงจรทางดิจิทัล การออกแบบวงจรให้แยกกราวด์ของสัญญาณอนาล็อกกับกราวด์ของสัญญาณดิจิทัลออกจากกัน เพื่อป้องกันปัญหาของสัญญาณรบกวน (noise) ซึ่งวิธีนี้จะช่วยลดสัญญาณรบกวนลงได้อย่างมาก

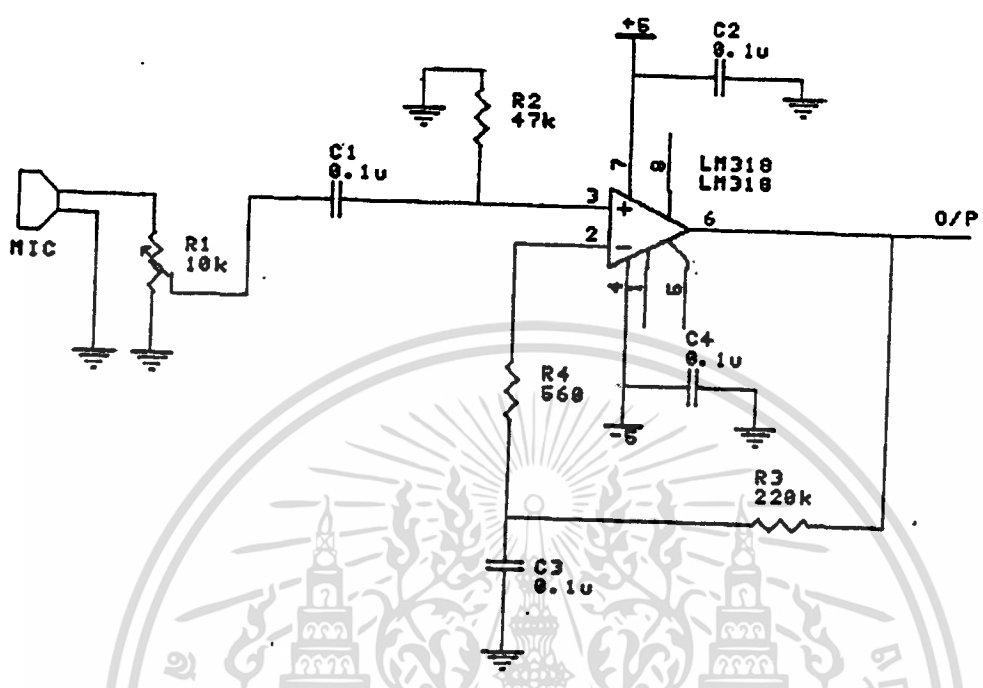
### 5.1 โครงสร้างของฮาร์ดแวร์



รูปที่ 5.2 แสดงบล็อกโคอะแกรมของฮาร์ดแวร์

สัญญาณอินพุทเมื่อเริ่มแรกเป็นสัญญาณเสียงจากไมโครโฟนนำมาขยายที่ preamp จากนั้นนำสัญญาณไปสู่ตัวกรองผ่านความถี่ต่ำ (low pass filter) จากนั้นนำไปกระดัยสัญญาณ เพื่อให้ได้สัญญาณอยู่ในช่วง 0-5 V ตามต้องการ

### วงจรมายาสัญญาณ



รูปที่ 5.3 วงจรมายาสัญญาณ

วงจรมายาสัญญาณจาก ไมโครโฟน โดยจากใช้ไอซีออปแอมป์ LM318 ทำหน้าที่เป็นวงจรมายาสัญญาณแบบไม่กลับเฟส (Non Inverting Amplifier) ตัวต้านทาน 47 กิโลโอห์ม ทำการจัดกระแสให้กับ non inverting amplifier input ตัวต้านทานและตัวเก็บประจุ 10 กิโลฟาร์ด เป็นตัวกำหนดความถี่คutoff ด้านต่ำประมาณ 30 เฮิร์ต ซึ่งคำนวณจาก

$$2f = 1/RC$$

$$2f = 1/(560 * 10)$$

$$= 28.42 \text{ เฮิร์ต}$$

อัตราการขยายของวงจรมายาสามารถหาได้จาก

$$V_0/V_1 = R_3 / \sqrt{(R_2)^2 + (1/C_2)^2}$$

ที่ความถี่ 30 เฮิร์ตซ์ คำนวณอัตราการขยายได้ดังนี้

$$V_0/V_1 = 220k / \sqrt{(560)^2 + (1/2 * \pi * 10\mu)^2}$$

$$= 285.2$$

$$= 20 \log(285.2) \text{ dB}$$

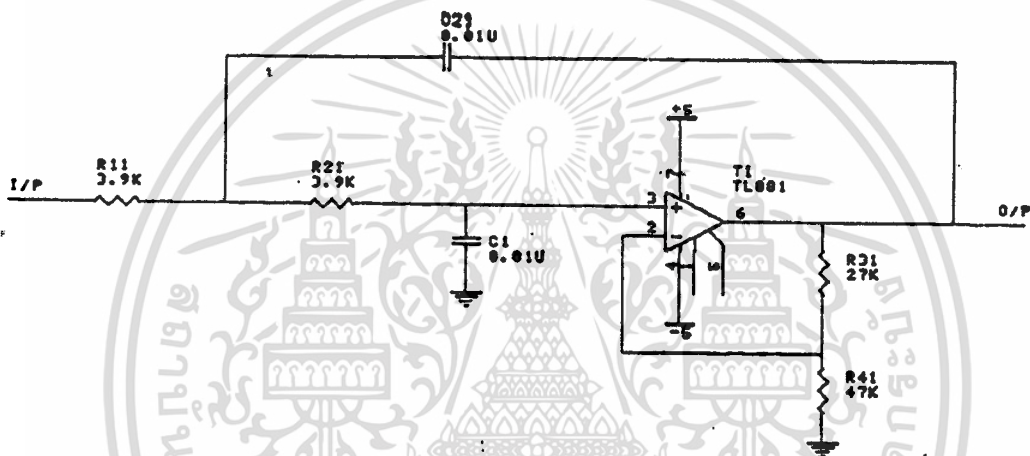
$$= 49.1 \text{ dB}$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ที่ความถี่เหนือ 30 เฮิรตซ์ หายจางหายไปดังนี้

$$\begin{aligned} V_0/V_1 &= R_3 / R_2 \\ &= 220k / 560 \\ &= 392.86 \\ &= 10 \log(392.86) \text{ dB} \\ &= 51.9 \text{ dB} \end{aligned}$$

วงจรกรองความถี่ต่ำผ่าน (low pass filter)

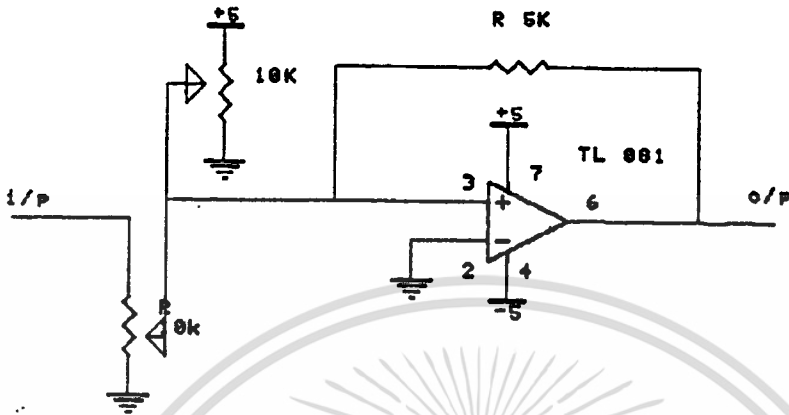


รูปที่ 5.4 วงจรกรองความถี่ต่ำผ่าน

วงจรกรองความถี่ต่ำผ่าน เป็นวงจรกรองความถี่ต่ำออร์เดอร์ที่ 2 (2<sup>nd</sup> order low pass filter) โดยใช้ไอซีเบอร์ LT081 ทำหน้าที่กรองสัญญาณโดยมีความถี่คัทออฟประมาณ 4 กิโลเฮิรตซ์ โดยใช้ตัวต้านทาน 3.9 กิโลโอห์ม และตัวเก็บประจุขนาด 0.01 ไมโครฟารัด สามารถคำนวณความถี่คัทออฟได้จาก

$$\begin{aligned} f_0 &= 1 / 2 * \pi * R * C \\ &= 1 / 2 * \pi * 3.9k * 10\mu \\ &= 4.08 \text{ กิโลเฮิรตซ์} \end{aligned}$$

วงจรรวบรวมระดับสัญญาณ (shifting amplifier)



รูปที่ 5.5 วงจรรวบรวมระดับสัญญาณ

วงจรรวบรวมระดับสัญญาณ ใช้เพื่อทำการปรับ สัญญาณอินพุตให้มีระดับอยู่ระหว่าง 0 ถึง 5 โวลต์ เพื่อป้องกันเข้ากับตัวแปลงสัญญาณ โดยใช้โอซีเบอร์ TL081 ทำหน้าที่รวบรวมระดับสัญญาณ ซึ่งมีตัวต้านทานที่เปลี่ยนค่าได้ขนาด 10 กิโลโอห์ม 2 ตัวเป็นตัวปรับระดับสัญญาณ

A/D ใช้สำหรับการเปลี่ยนสัญญาณอนาลอกเป็นสัญญาณดิจิทัล โดยใช้โอซีเบอร์ ADC 0808 ซึ่งตัวโอซีนี้ใช้สัญญาณคล็อกจาก external clock ที่มีความถี่ 10 กิโลเฮิร์ต ซึ่งตั้งค่าโวลต์ตรงอ้างอิงที่ 0-5 โวลต์ ขาสัญญาณ START กับ BOC ต่อเข้าด้วยกันเพื่อใช้ในโหมด free running สัญญาณนาฬิกาที่ป้อนให้กับ ADC0808 ใช้สัญญาณนาฬิกาที่มีความถี่ 1.25 เมกะเฮิร์ต แล้วจะได้ conversion times เท่ากับ 0.1 มิลลิวินาที หรืออัตราการสุ่มสัญญาณเป็น 10 กิโลเฮิร์ต

external clock เป็นการนำสัญญาณ 20 เมกะเฮิร์ตจากตัวคริสตัลมาหารสัญญาณ เพื่อจะนำสัญญาณนาฬิกา 1.25 เมกะเฮิร์ตมาใช้งานป้อนให้แก่ analog to Digital converter ซึ่งสัญญาณดังกล่าวได้จาก clock generator 8284 และใช้คริสตัลสัญญาณนาฬิกา 20 เมกะเฮิร์ตกำกับสัญญาณอีกทีหนึ่ง แล้วนำมาหารความถี่ด้วย 74LS390 3 ตัวก็จะได้สัญญาณนาฬิกา 1.25 เมกะเฮิร์ต

counter address ทำหน้าที่อ้างอิงตำแหน่งของหน่วยความจำ (external ram) ตามจังหวะสัญญาณนาฬิกาจากสัญญาณนาฬิกาภายในฮาร์ดแวร์ หรือสัญญาณนาฬิกาจากคอมพิวเตอร์อย่างใดอย่างหนึ่ง ถ้าอยู่ในช่วงแปลงข้อมูลจากอนาลอกเป็นดิจิทัล แล้วเขียนลงหน่วยความจำก็จะใช้สัญญาณนาฬิกาภายในฮาร์ดแวร์

แต่ถ้าเป็นการอ่านข้อมูลจากหน่วยความจำเข้าสู่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น เมื่อนำไปใช้โดยไม่แจ้งการคัดค้านการคัด

ไม่ว่าการณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

คอมพิวเตอร์ก็จะขึ้นกับสัญญาณนาฬิกาจากคอมพิวเตอร์ ทั้งนี้เพื่อให้จังหวะการทำงานของการเขียนข้อมูลลงหน่วยความจำตรงกับอัตราการสุ่มข้อมูลและจังหวะการอ่านข้อมูลจากหน่วยความจำตรงกับกรเรียกข้อมูล (inport) เข้าสู่คอมพิวเตอร์ โดยใช้ไอซีเบอร์ 74LS393 4 ตัวในการนับตำแหน่ง

external RAM เป็น RAM แบบสะแตติก ขนาด 32 k X 8 บิต ซึ่งมีเนื้อที่หน่วยความจำ 32 k ตำแหน่ง ตำแหน่งละ 8 บิต แต่ใช้อย่างอิงในการวิจัยเพียง 20 k ตำแหน่ง โดยทำหน้าที่เก็บข้อมูลสัญญาณเสียงซึ่งเปลี่ยนเป็นสัญญาณดิจิทัลแล้ว

8255 ทำหน้าที่เป็นพอร์ตขนานสำหรับติดต่อกับคอมพิวเตอร์ การทำงานของ 8255 จะทำการควบคุมจากคอมพิวเตอร์ โดยส่งคำสั่งมาโปรแกรมการทำงานหรือกำหนดรูปแบบของพอร์ต การเรียกพอร์ตของ 8255 จะเรียก พอร์ตต่าง ๆ ว่า พอร์ต A พอร์ต B และ พอร์ต C โดยพอร์ต C แยกเป็นสองส่วนคือ พอร์ต C ล่าง และ พอร์ต C บน

D0-D7	ขาข้อมูล (data bus) แบบ 2 ทิศทาง
A0-A1	ใช้งานร่วมกับสัญญาณ RD และ WR เพื่อเลือกและควบคุม 1 ใน 3 พอร์ต หรือคอนโทรลเวิร์ดรีจิสเตอร์ (control word register) PA PB PC พอร์ตทั้ง 3 ของ 8255
CS	(chip select) สัญญาณเลือกชิพเพื่อใช้งาน
RD	(read) เป็นสัญญาณการอ่านข้อมูลภายใน 8255
WR	(write) เป็นสัญญาณในการเขียนข้อมูลลงใน 8255
RESET	เป็นสัญญาณรีเซ็ตแก่ 8255 เพื่อเคลียร์รีจิสเตอร์

เราควบคุมการทำงานภายในของ 8255 โดยส่งสัญญาณไปโปรแกรมพอร์ตควบคุม (control port) ซึ่งจะต้องมีการโปรแกรมในคอนแรกก่อนการใช้งาน การโปรแกรม 8255 แบ่งออกเป็น 3 โหมด คือ โหมด 0 โหมด 1 และ โหมด 2 สำหรับการใช้งานฮาร์ดแวร์จะใช้โหมด 0 ซึ่งในการวิจัยนี้ใช้ control word ดังนี้ 1001001

ความหมายของบิตต่างๆ มีดังนี้

- บิต D7 เป็นบิตที่แสดงรหัสควบคุม ถ้าบิตนี้เป็น "1" หมายถึง รหัสควบคุมนี้จะมีผลต่อการเปลี่ยนแปลงการเซตโหมดต่างๆ ของ 8255
- บิต D6 และ D5 เป็นการเลือกโหมดของพอร์ต A ซึ่งมี 3 โหมด คือ โหมด 0, 1 และ 2
- บิต D4 ถ้ามีค่าเป็น "0" หมายถึง พอร์ต A เป็น พอร์ตเอาต์พุต ถ้ามีค่าเป็น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับ "1" หมายถึง กำหนดพอร์ต A เป็นพอร์ตเอาต์พุต ใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- บิต D3 เป็นบิตที่บอกถึงการเชื่อมต่อพอร์ต C บน ถ้าเป็น "0" จะทำให้เป็น  
เอาต์พุต
- บิต D2 เป็นบิตที่เป็นการเชื่อมต่อของพอร์ต B ถ้าเป็น "0" หมายถึง  
พอร์ต B เป็นโหมด 0 แต่ถ้าเป็น "1" หมายถึง การเลือกโหมด 1
- บิต D1 เป็นการกำหนดการเลือกอินพุตเอาต์พุตของพอร์ต B ถ้าเป็น "0"  
หมายถึง เอาต์พุต ถ้าเป็น "1" หมายถึง อินพุต

clock computer เป็นจังหวะการนับสัญญาณตามคำสั่งในโปรแกรมโดยผ่านทาง  
พอร์ต C บน

computer การเชื่อมต่อระหว่างคอมพิวเตอร์กับฮาร์ดแวร์ จะทำโดยผ่านตัว 8255  
ซึ่งการใช้งานจะต้องกำหนดแอดเดรสขึ้นมาเพื่อที่จะสามารถอ้างอิงได้โดยโปรแกรมแต่สำหรับ  
แอดเดรสค่าต่างๆ จะมีกำหนดไว้ให้อุปกรณ์อื่นใช้งานอยู่แล้วดังตารางที่ 6.1

สำหรับการเลือกใช้แอดเดรสสำหรับฮาร์ดแวร์นี้ จะเลือกใช้แอดเดรสที่อยู่ในช่วง  
300-31F ซึ่งเป็นแอดเดรสที่เอาไว้สำหรับการคัดค้นแบบ(prototype card)จึงสามารถใช้ค่าแอดเดรส  
ในช่วงนี้ได้ แต่ต้องระวังมิให้มีการอ้างแอดเดรสซ้ำซ้อนในกรณีที่มีการคัดค้นแบบมากกว่าหนึ่ง  
การคัดค้นไป

แอดเดรสที่ใช้งานมีค่าคือ 31C 31D 31E และ 31F จำนวน 4 แอดเดรสเพื่อใช้งาน  
โดยเรียกผ่าน 8255 อีกที

แอดเดรส 31C ใช้เป็นพอร์ตอินพุต สำหรับรับข้อมูลเสียงที่ผสมปลิง  
เข้ามาซึ่งตรงกับพอร์ต A ของ 8255

แอดเดรส 31D ใช้เป็นพอร์ตเอาต์พุตสำหรับส่งข้อมูลเสียง ไปแสดงผล  
ทางลำโพงซึ่งตรงกับพอร์ต B ของ 8255

แอดเดรส 31E ใช้เป็นพอร์ตควบคุม คอมพิวเตอร์จะส่งและรับสัญญาณ  
ควบคุมต่างๆ ทางพอร์ตนี้ ได้แก่ สัญญาณอ่าน/เขียน  
ข้อมูล สัญญาณนาฬิกาจากคอมพิวเตอร์ สัญญาณเคลียร์  
เคาท์เตอร์ สัญญาณสิ้นสุดการอ่าน/เขียนหน่วยความจำ  
ซึ่งตรงกับพอร์ต C ของ 8255 โดยพอร์ต C บน เป็น  
เอาต์พุตพอร์ตที่ส่งสัญญาณควบคุมออกจากคอมพิวเตอร์  
และพอร์ต C ล่าง เป็นอินพุตที่รับสัญญาณสิ้นสุดการ  
อ่าน/เขียน หน่วยความจำเข้าคอมพิวเตอร์

แอดเดรส 31F ใช้ส่งสัญญาณเพื่อที่จะโปรแกรมการใช้งาน 8255

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับใช้ในงานเพื่อการศึกษาเท่านั้น มิอนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

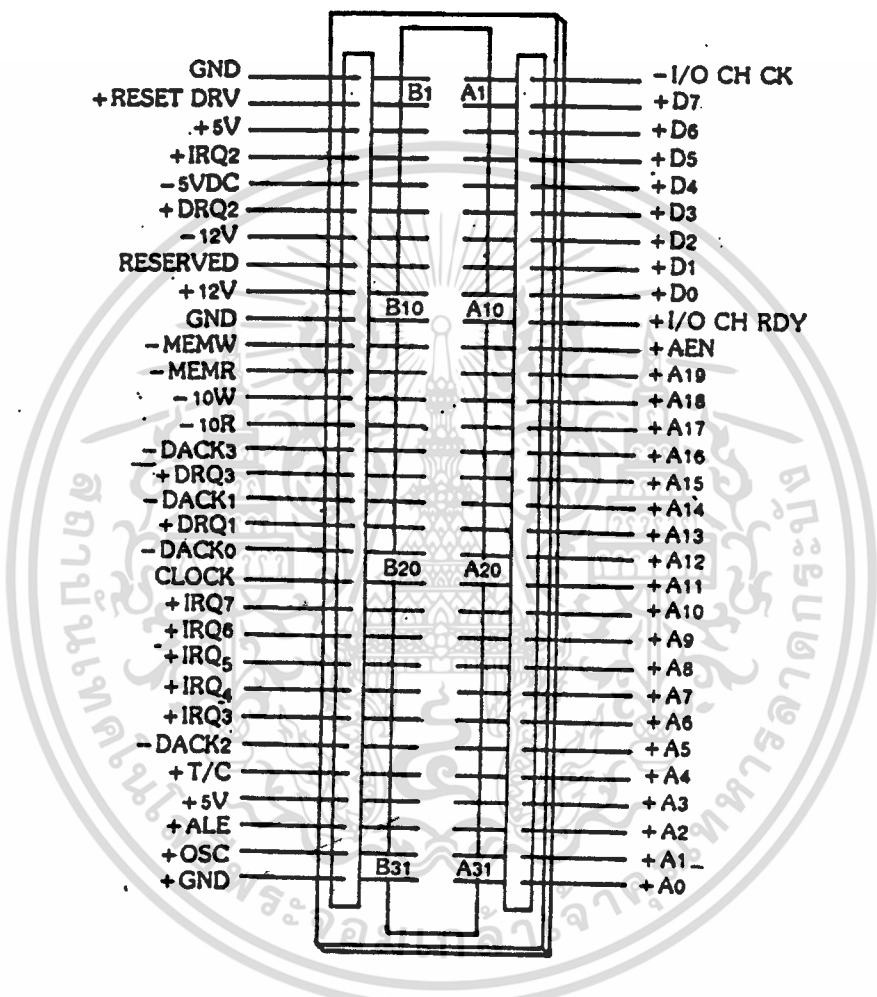
Hex Range	Usage
00000F	DMA Chip 8237A-5
020-021	Interrupt 8259A
040-043	Timer 8253-5
060-063	PPI 8255A-5
080-083	DMA Page Registers
0AX	NMI Mask Registers
0CX	Reserved
0EX	Reserved
200-20F	Game Control
210-217	Expansion Unit
220-24F	Reserved
278-27F	Reserved
2F0-2F7	Reserved
2F8-2FF	Asynchronous Communications (See)
300-31F	Prototype Card
320-32F	Fixed Disk
378-37F	Printer
380-38C	SDLC Communications
380-389	Binary Synchronous Communications (See)
3A0-3A9	Binary Synchronous Communications (See)
3B0-3BF	IBM Monochrome Display/Printer
3C0-3CF	Reserved
3D0-3DF	Color/Graphics
3E0-3E7	Reserved
3F0-3F7	Diskette
3F8-3FF	Asynchronous Communications (Pri)

เอกสารนี้เป็นเอกสารร่างที่ 5.1 การใช้งานแอดแดรสสำหรับพอร์ตอินพุทเอาต์พุทต่างๆ ใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การเชื่อมต่อทางฮาร์ดแวร์ต้องคำนึงถึงสัญญาณบนสล็อตคอมพิวเตอร์ด้วย ซึ่งมีรายละเอียดดังนี้

การเชื่อมต่อกับคอมพิวเตอร์ เนื่องจากฮาร์ดแวร์เป็นอุปกรณ์ประเภทอินพุท เอาท์พุท จะต้องเชื่อมต่อกับคอมพิวเตอร์ผ่านทางอินพุทเอาท์พุทพอร์ต (I/O port)

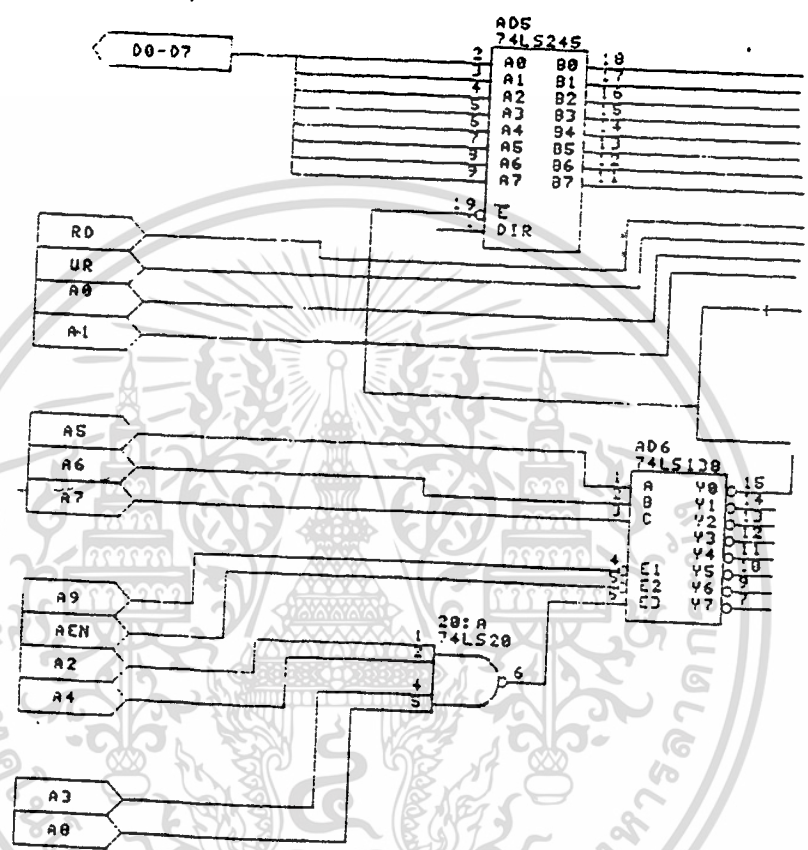


รูปที่ 5.6 แสดงขาสัญญาณต่างๆ บนสล็อตคอมพิวเตอร์

เส้นสัญญาณต่างๆ ประกอบด้วยเส้นสัญญาณของบัสแอดเดรส (address bus) บัสข้อมูล (data bus) บัสควบคุมสำหรับการเขียน/อ่านข้อมูลจากหน่วยความจำ หรือ พอร์ตอินพุทเอาท์พุท เส้นสัญญาณสำหรับการขออินเตอร์รัพท์ของวงจรรีเฟรชอินเตอร์เฟส เส้นสัญญาณสำหรับการขอ DMA สัญญาณฐานเวลา (timing signal) ต่างๆ ที่ใช้ในระบบเส้นสัญญาณแสดงการรีเฟรชหน่วยความจำ และสัญญาณสำหรับการตรวจสอบความผิดพลาด (I/O CHECK)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

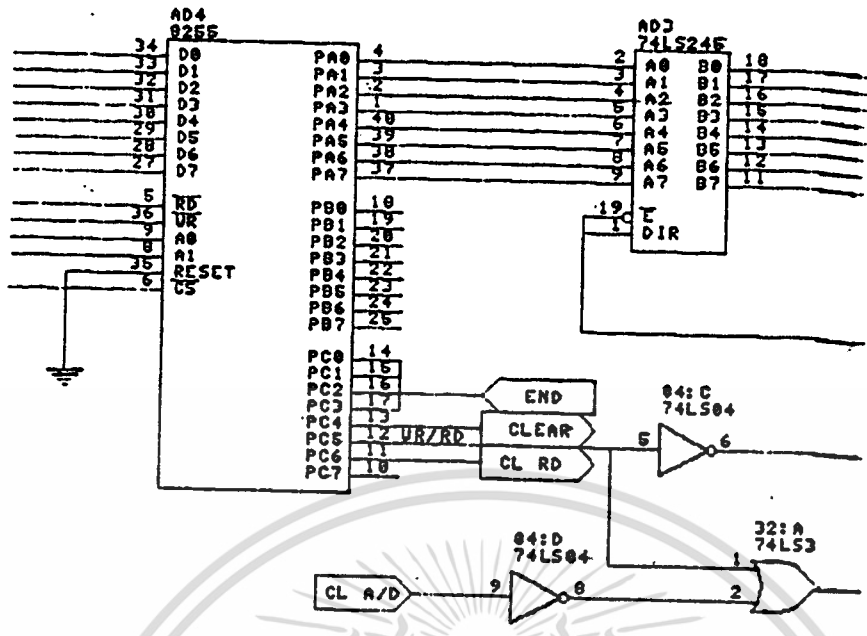
นอกจากเส้นสัญญาณเหล่านี้ สล็อตบนเมนบอร์ดยังคํอกับแหล่งจ่ายไฟที่ใช้ใน ระบบ  
อีก คือ +5 Vdc, -5 Vdc, +12 Vdc และ -12 Vdc



รูปที่ 5.7 วงจร INTERFACE

จากรูปวงจร สัญญาณแอกเครสบ้สจากคอมพิวเตอร์จะมาเข้า 74LS20 และ 74LS138 เพื่อทำการตีโค้ดการเรียกใช้งานของฮาร์ดแวร์ให้สามารถติดต่อใช้งานได้ในช่วงแอกเครส 31C-31F สำหรับสัญญาณจากคาส์บัสจะผ่านบัฟเฟอร์ 74LS245 เข้าสู่ 8255 เพื่อแยกเป็น 3 พอร์ตไว้ติดต่ออินพุทเอาท์พุทกับส่วนต่างๆ ของฮาร์ดแวร์ โดยขาสัญญาณที่ 15 จะทำหน้าที่ส่งสัญญาณแก่ ขา CS ของ 8255 และ ขา 19 ของ BUFFER เพื่อทำ การส่งสัญญาณจากคอมพิวเตอร์ผ่านทาง 8255 สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นอนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 5.8 วงจร 8255 และ BUFFER

จากวงจร 8255 พอร์ต A เป็นพอร์ตอินพุต อินพุตสัญญาณจากการรุ่มตัวอย่าง เข้าสู่คอมพิวเตอร์โดยผ่านทาง BUFFER (74LS245) โดยจะมีสัญญาณกระตุ้นจากสัญญาณ WR/RD ซึ่งจะมาจาก พอร์ต C

พอร์ต B เป็นเอาต์พุตพอร์ต สำหรับส่งข้อมูลจากคอมพิวเตอร์ผ่านวงจรแปลงสัญญาณดิจิทัลเป็นอนาลอก แล้วขยายสัญญาณออกสู่โพง

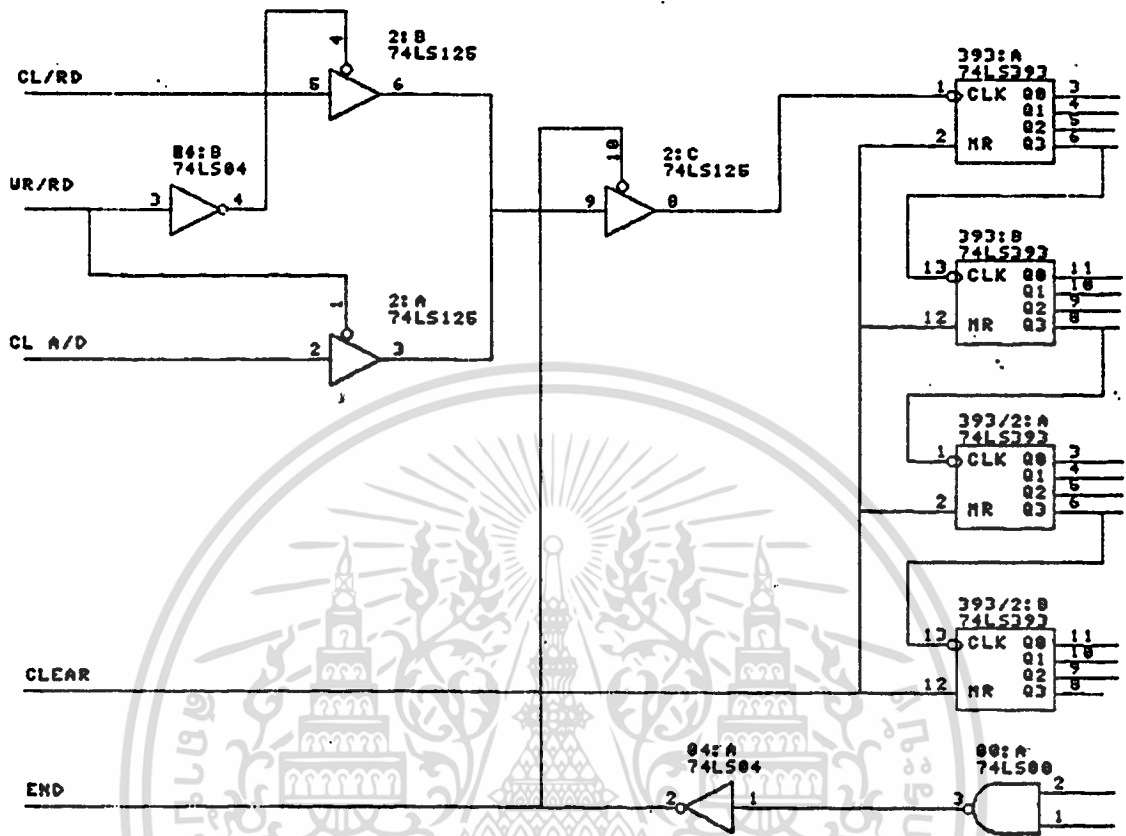
พอร์ต C เป็นเอาต์พุตพอร์ต จะส่งสัญญาณจากคอมพิวเตอร์มาควบคุมส่วนต่างๆ ของฮาร์ดแวร์ 3 สัญญาณ คือ

- สัญญาณ CLEAR จะใช้สำหรับการรีเซ็ตเคาท์เตอร์แอดเดรสเพื่อเริ่มต้นนับแอดเดรสหน่วยความจำใหม่ในการอ่านและเขียนข้อมูลแต่ละครั้ง
- สัญญาณ WR/RD เป็นสัญญาณที่แสดงถึงช่วงเวลาการเขียนหรืออ่าน การเขียนคือการเขียนข้อมูลที่ได้จากการรุ่มตัวอย่างลงหน่วยความจำ การอ่านคือการอ่านข้อมูลที่เก็บในหน่วยความจำในการอ่านข้อมูล
- สัญญาณ CL/RD เป็นสัญญาณนาฬิกาจากคอมพิวเตอร์ที่ใช้ในการอ้างแอดเดรสหน่วยความจำในการอ่านข้อมูล

พอร์ต C ล่าง เป็นอินพุตพอร์ตจะรับสัญญาณ END จากฮาร์ดแวร์สัญญาณ END เป็นสัญญาณที่บอกถึงการสิ้นสุดการอ่าน/เขียนหน่วยความจำ เพื่อให้คอมพิวเตอร์ทำงานอื่นต่อไป

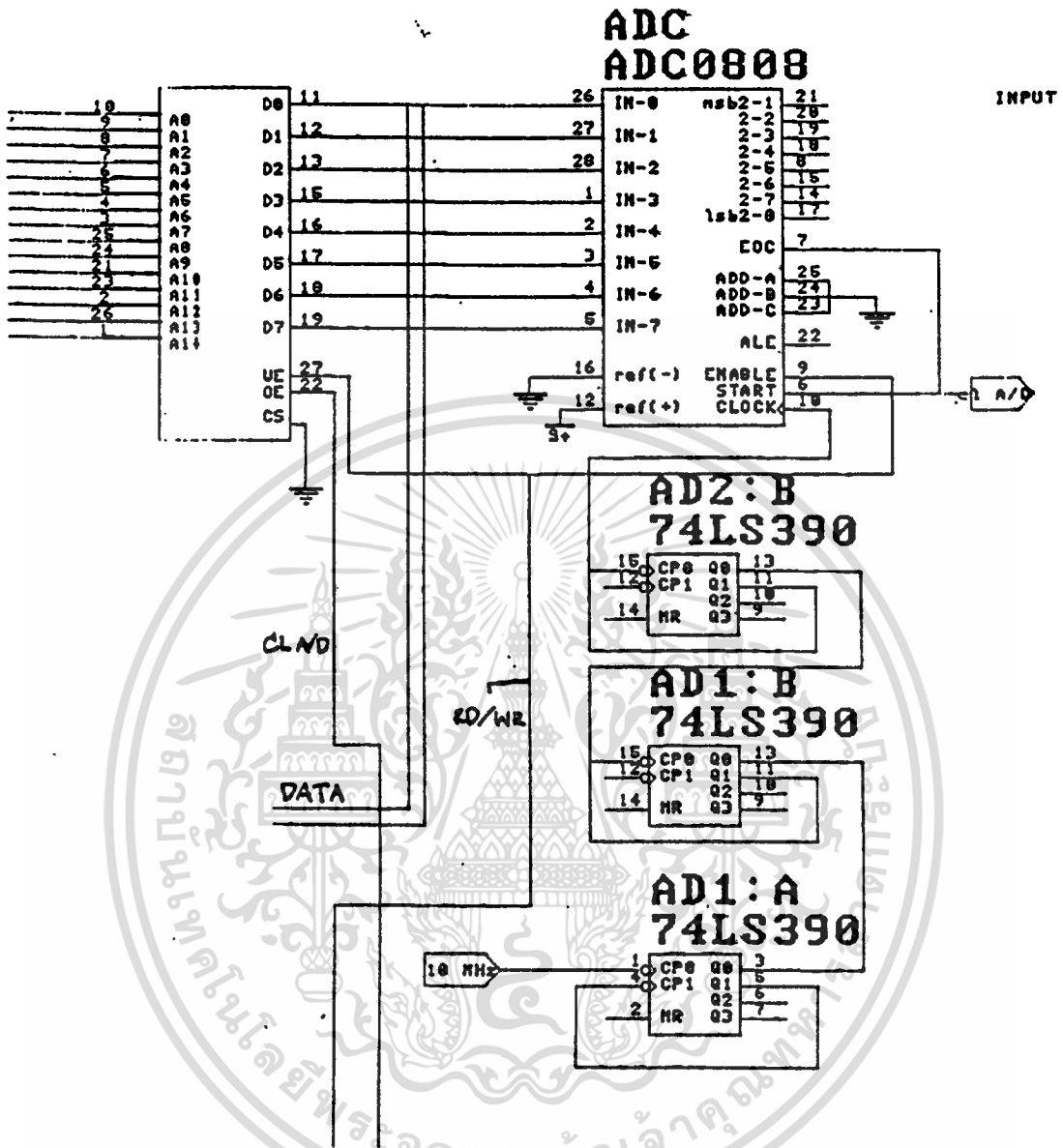
เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ไว้กับการใช้งานเพื่อการศึกษาเท่านั้น ไม่สามารถนำออกจำหน่าย

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 5.9 วงจร นับตำแหน่ง

74LS125 ทั้ง 3 ตัวทำหน้าที่คล้ายสวิทซ์ในการเลือกสัญญาณ WR/RD เป็นสัญญาณอ้างอิง สัญญาณนาฬิกาอ้างอิงการอ่านหรือเขียนข้อมูลผ่านทางไมโครโปรเซสเซอร์ 74LS393 จากรูป จะใช้ 74LS393 จำนวน 4 ตัว เพื่อนับให้ได้ถึง 15 บิต หรือ สามารถอ้างอิงหน่วยความจำจำนวน 20 กิโลไบต์ ได้ เมื่อเคาท์เตอร์นับถึง 20 กิโลไบต์ โดยมีสัญญาณ WR/RD ส่งจากพอร์ต C บนภาควควบคุมการนับตำแหน่งของสัญญาณ แล้วจะผ่าน 74LS00 และ 74LS04 เพื่อไปเป็นสัญญาณ END และไปใช้ในการเอเบิล 74LS125 ตัวที่ 3 ด้วย เพื่อตัดสัญญาณ และทำหน้าที่แยกคล็อกของฮาร์ดแวร์กับคล็อกของคอมพิวเตอร์เพื่อให้เกิดการซิงค์ไครโนซ์



รูปที่ 5.10 วงจร A/D , วงจร หารสัญญาณ และ วงจรหน่วยความจำ

หน่วยความจำจะใช้ไอซีเบอร์ 62256 ซึ่งเป็นหน่วยความจำขนาด 32K x 8 บิต คือมี  
 เนื้อที่หน่วยความจำ 32K ตำแหน่ง ตำแหน่งละ 8 บิต แต่ที่ใช้ในวงจรนี้จะอ้างอิงแต่จำนวน 20 K  
 เท่านั้น โดยได้รับสัญญาณคล็อกจาก A/D โดยจะได้สัญญาณ WR/RD จาก 8255 พอร์ต C บน  
 ซึ่งจะผ่านมาทาง 74LS32 และ 74LS04

74LS390 จะทำหน้าที่เป็นควอเตอร์สัญญาณนาฬิกา 10 เมกะเฮิร์ต ให้ได้ 1.25  
 เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้เผยแพร่ไปใช้  
 ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมกะเฮิร์ตเพื่อป้อนให้กับ A/D

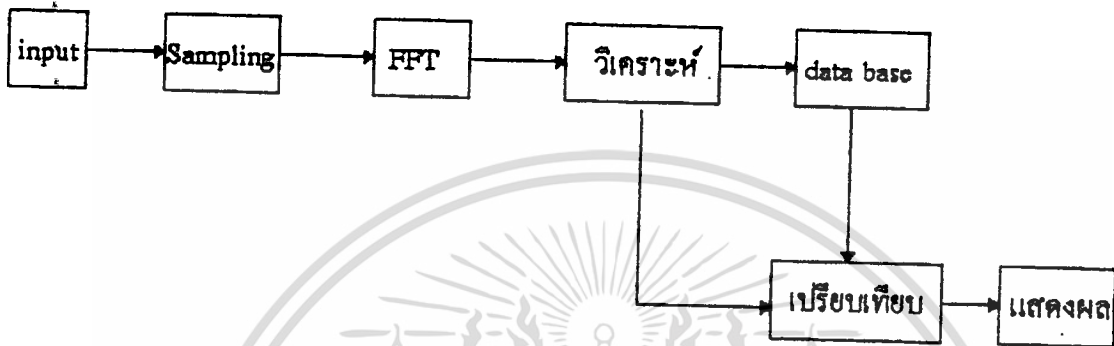
A/D ใช้สำหรับการเปลี่ยนสัญญาณอนาลอกเป็นสัญญาณดิจิทัล โดยใช้ไอซีเบอร์ ADC 0808 ซึ่งตั้งค่าโวลต์แดงอ้างอิงที่ 0-5 โวลต์ ขาสัญญาณ START กับ EOC คอ่เข้าด้วยกัน เพื่อใช้ในโหมด free running สัญญาณนาฬิกาที่ป้อนให้กับ ADC0808 ใช้สัญญาณนาฬิกาที่ ความถี่ 1.25 เมกะเฮิร์ต แล้วจะได้ conversion times เท่ากับ 0.1 มิลลิเซค หรืออัตราการสุ่ม สัญญาณเป็น 10 กิโลเฮิร์ต



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## บทที่ ๘ ทฤษฎีที่เกี่ยวข้องกับงานวิจัย

การวิเคราะห์เปรียบเทียบสัญญาณเสียงแสดงดังรูปที่ 6.1



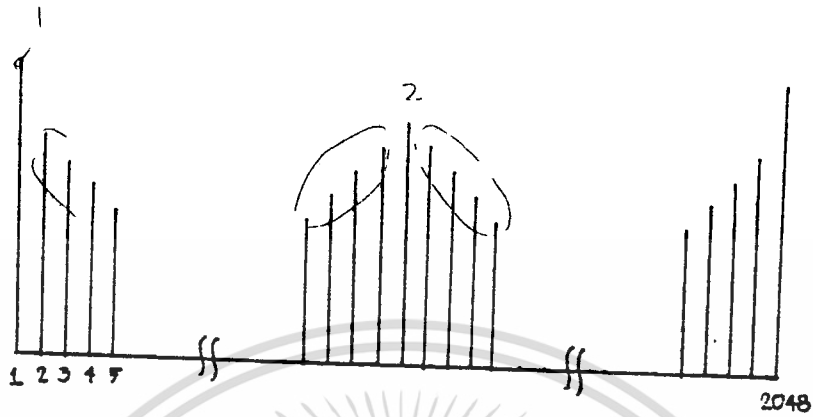
รูปที่ 6.1 แสดงบล็อก ไดอะแกรมของการวิเคราะห์สัญญาณเสียง

-input เป็นสัญญาณเสียงที่ผ่านการแปลงอนาลอกมาเป็นดิจิทัลโดยฮาร์ดแวร์แล้ว เพื่อทำการประมวลผลต่อไป

-Sampling เป็นการนำอินพุตมาสุ่มหาตัวอย่างโดยวิธีคว้นไคท์ ใช้หลักการของอัตราในควิส์

-FFT นำสัญญาณที่ได้จากการ sampling มาทำการแปลงฟูริเยร์เพื่อหาฮาร์โมนิกของสัญญาณเสียง โดยใช้วิธีคทอนทางเวลา ซึ่งขนาดของ  $N$  มีค่าเท่ากับ 2048 จุด

-การวิเคราะห์ เป็นการนำสัญญาณซึ่งผ่านกระบวนการ FFT แล้วมาทำการวิเคราะห์หาค่าเฉลี่ยของฮาร์โมนิกหลักๆ 8 ตัว ซึ่งแบ่งออกเป็น 2 ส่วน คือ ชุดของฮาร์โมนิกตัวที่ 1 ถึง ฮาร์โมนิกตัวที่ 1024 และ ชุดของฮาร์โมนิกตัวที่ 1025 ถึง ฮาร์โมนิกตัวที่ 2048 หลังจากได้ตัวฮาร์โมนิกหลักมาแล้วจะต้องทำการเก็บฮาร์โมนิกตัวข้างเคียงของตัวหลักอีก 8 ตัว



รูปที่ 6.2 แสดงฮารโมนิกของเสียง

ฮารโมนิกตัวที่ 1 ซึ่งจะเป็นฮารโมนิกตัวที่มีค่ามากที่สุดและเก็บฮารโมนิกตัวข้างเคียงกับฮารโมนิกตัวที่ 1 คือ ฮารโมนิกตัวที่ 2 ถึง ฮารโมนิกตัวที่ 5 เป็นจำนวน 4 ตัว

ฮารโมนิกอันดับสอง ที่จะเก็บจะเป็นฮารโมนิกตัวที่มีค่ารองจากฮารโมนิกตัวที่ 1 ซึ่งฮารโมนิกตัวนี้จะต้องเป็นฮารโมนิกที่อยู่ระหว่างฮารโมนิกที่ 1-1024 เมื่อได้ฮารโมนิกอันดับ 2 แล้วก็เก็บฮารโมนิกตัวข้างเคียงอีก 8 ตัว ทั้งทางซ้ายและขวา

ฮารโมนิกอันดับที่ 3 จะเป็นฮารโมนิกที่มีค่าอันดับที่ 3 จากฮารโมนิกตัวที่ 1 และเก็บฮารโมนิกข้างเคียงซ้ายขวาอีก 8 ตัว

ฮารโมนิกอันดับที่ 4 คือ ฮารโมนิกที่มีค่าอันดับที่ 4 รองจากฮารโมนิกตัวที่ 1 ซึ่งจะ เป็นฮารโมนิกตัวสุดท้ายของชุดแรก (ฮารโมนิกตัวที่ 1 ถึงฮารโมนิกตัวที่ 1024) และเก็บฮารโมนิกตัวข้างเคียงทั้งซ้ายขวาอีก 8 ตัว

ฮารโมนิกที่สำคัญอีกตัวที่ต้องทำการเก็บคือ ฮารโมนิกตัวที่ 2048 ซึ่งมีความสำคัญพอๆ กับฮารโมนิกตัวที่ 1 และเก็บฮารโมนิกข้างเคียงคือ ฮารโมนิกตัวที่ 2044 ถึง ฮารโมนิกตัวที่ 2047 เป็นจำนวน 4 ตัว

ฮารโมนิกถัดจากฮารโมนิกตัวที่ 2048 ที่ต้องเก็บคือ ฮารโมนิกตัวที่มีค่ารองจากฮารโมนิกตัวที่ 2048 ซึ่งต้องอยู่ระหว่างฮารโมนิกตัวที่ 1025 ถึงฮารโมนิกตัวที่ 2045 และเก็บฮารโมนิกตัวข้างเคียงซ้ายขวาอีก 8 ตัว

ฮารโมนิกอันดับที่ 7 จะเก็บเป็นฮารโมนิกที่มีค่าสูงเป็นอันดับ 3 เมื่อเปรียบเทียบกัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาค้นคว้าเท่านั้น ไม่ควรนำออกเผยแพร่โดยไม่ได้รับอนุญาต  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ฮาร์โมนิคตัวที่ 2048 และเก็บฮาร์โมนิคตัวข้างเคียงซ้ายขวาอีก 8 ตัว

ฮาร์โมนิคหลักตัวสุดท้าย ที่จะต้องทำการเก็บคือ ฮาร์โมนิคที่มีร่าอันดับที่ 4 รองจากฮาร์โมนิคตัวที่ 2048 และเก็บฮาร์โมนิคซ้ายขวาอีก 8 ตัว

เมื่อทำการวิเคราะห์หาค่าเฉลี่ยของฮาร์โมนิคหลัก และฮาร์โมนิคข้างเคียงทุกตัวแล้ว (ซึ่งค่าเฉลี่ยที่ทำการหามาจากการนำค่าฮาร์โมนิคของเสียงพูดที่เป็นฐานเสียงมาหาค่าเฉลี่ยผลคูณคิด เช่น หาค่าเฉลี่ยของฮาร์โมนิคตัวที่ 1 โดยนำค่าของฮาร์โมนิคตัวที่ 1 ของเสียงที่พูดเป็นฐานเสียงทุกครั้งมาหาค่าเฉลี่ย) ต่อจากนั้นนำค่าเฉลี่ยมาหาค่าเบี่ยงเบนมาตรฐานเพื่อจะได้นำมาเก็บเป็นฐานข้อมูลเพื่อการเปรียบเทียบต่อไป

- ค่าค่าเบส จะเก็บค่าจริง ค่าเฉลี่ยของ ค่าเบี่ยงเบนมาตรฐาน ค่าเฉลี่ยของค่าเบี่ยงเบนมาตรฐาน ของข้อมูลฮาร์โมนิคตัวหลักและฮาร์โมนิครองตามที่กล่าวมาแล้ว โดยจะทำการเก็บแยกข้อมูลต่างๆ ไว้เป็นไฟล์ เพื่อความสะดวกในการเข้าถึงข้อมูล

- ไฟล์ของค่าที่แท้จริงของฮาร์โมนิคตัวหลัก เช่น ไฟล์ค่าจริงของฮาร์โมนิคตัวที่ 1 จะเก็บค่าของฮาร์โมนิคตัวที่ 1 ของเสียงทุกครั้งที่มีการเก็บเสียงไว้เป็นฐานเสียง

- ไฟล์เก็บค่าเฉลี่ยของฮาร์โมนิค เช่น ไฟล์ค่าเฉลี่ยของฮาร์โมนิคตัวที่ 2048 จะเก็บค่าเฉลี่ยของฮาร์โมนิคตัวที่ 2048 ซึ่งได้จากการคำนวณหาค่าเฉลี่ยแล้ว

- ไฟล์เก็บค่าเบี่ยงเบนมาตรฐาน

- ไฟล์เก็บค่าเฉลี่ยของค่าเบี่ยงเบนมาตรฐาน

- การเปรียบเทียบ เมื่อมีคำสั่งเพื่อการเปรียบเทียบเสียง จะมีการนำเสียงที่พูดเข้ามาผ่านกระบวนการ sampling FFT และการวิเคราะห์ เพื่อเปรียบเทียบกับค่าที่เก็บไว้เป็นค่าค่าเบสของค่าที่มีอยู่

- แสดงผล เมื่อทำการเปรียบเทียบแล้วก็จะแสดงผลทางหน้าจอคอมพิวเตอร์

Word dir is correct \_ %

Word one is correct \_ %

Word scan is correct \_ %

Word ohk is correct \_ %

Word command is \_ ซึ่งเป็นค่าที่มีค่าเปอร์เซ็นต์สูงสุด

## ทฤษฎีของการสุ่มสัญญาณ

สัญญาณที่ถูกนำเข้ามาผ่านระบบทำการสุ่มบางทีก็มีความหมายเป็นลำดับ (sequence) แล้วแต่ว่าจะเป็นลำดับเข้าหรือลำดับออก โดยธรรมชาติแล้วสัญญาณใดๆนั้นจะมีความยาวเป็นอนันต์ ดังนั้นเราจึงใช้เวกเตอร์เป็นตัวแทนข้อมูลให้มีความยาวที่แน่นอน ซึ่งจะทำการวิเคราะห์นั้น สามารถทำได้ง่ายขึ้นโดยการคำนวณด้วยระบบดิจิทัล

จากทฤษฎีดังกล่าว ทำให้เราสามารถอ้างได้ว่าข้อมูลเชิงตัวเลขที่เราสุ่มมาเพื่อเป็นตัวแทนสัญญาณทางอนาล็อกต่างๆนั้นย่อมมีจุดที่สุ่มที่แน่นอน หรือเราสามารถใช้อัตราการสุ่มข้อมูลเชิงตัวเลขต่างๆทางอนาล็อกได้เสมอและมีจำนวนที่แน่นอน เราเรียกความแตกต่างระหว่างข้อมูลทางตัวเลขที่เราสามารถกำหนดได้นี้ว่าข้อมูลทางควันไทซ์

การสุ่มข้อมูลแบบนี้อาจจะเกิดความคลาดเคลื่อนอยู่บ้าง เราเรียกว่าความผิดพลาดทางควันไทซ์ (quantizing error) ซึ่งความผิดพลาดทางควันไทซ์นี้เป็นปัญหาในเรื่องของสัญญาณรบกวน ซึ่งเป็นเรื่องปกติสำหรับการทำงานในระบบดิจิทัล และมักก่อให้เกิดความไม่เที่ยงตรงต่อระบบที่ทำงานเกี่ยวกับสัญญาณดิจิทัลนี้

ในระบบที่เรากำลังศึกษานี้ จะใช้ช่วงเวลาในการสุ่มข้อมูลแต่ละครั้งเท่ากันเสมอ ดังนั้นอัตราการสุ่มข้อมูลของระบบจะมีค่าเท่ากับส่วนกลับของเวลาที่แตกต่างกันระหว่างการสุ่มข้อมูลแต่ละครั้ง

ในการสุ่มข้อมูลจากสัญญาณทางอนาล็อกใดๆ ควรจะประมาณอัตราการสุ่มไว้ให้สูงพอสมควร เพราะถ้าใช้อัตราการสุ่มที่ค่อนข้างต่ำจนเกินไป อาจจะทำให้เกิดปัญหาในระหว่างที่ทำการสุ่มข้อมูลจากสัญญาณต่อเนื่องขึ้นได้ เพราะในสัญญาณนั้นๆอาจมีข้อมูลที่สำคัญมาก และอาจก่อให้เกิดการเหลื่อมล้ำของข้อมูลที่ทำการสุ่มขึ้นมาได้ ผลที่ได้คือเมื่อทำการแปลงข้อมูลที่สุ่มมาได้กลับไปในรูปแบบของสัญญาณทางอนาล็อก แล้วจะได้สัญญาณที่มีความผิดเพี้ยนจากสัญญาณต้นฉบับเดิมมาก แต่ถ้าเราใช้อัตราการสุ่มที่สูงเกินไป ซึ่งอาจจะมากเกินไปจนเกินไป จะทำให้เกิดความสิ้นเปลืองเนื้อที่ของหน่วยความจำโดยใช่เหตุ จะเห็นได้ว่าการสุ่มที่มากเกินไปก็ไม่ใช่วิธีที่ดีนัก

ทฤษฎีของการสุ่มที่ว่าด้วยค่าของอัตราในควิตซ์ซึ่งเป็นค่าการสุ่มข้อมูลแบบต่อเนื่องที่กล่าวถึงอัตราการสุ่มที่ต้องมีค่าน้อยที่สุดที่ยอมรับได้โดยไม่ทำให้เกิดการผิดเพี้ยนของข้อมูลอย่างน้อย 2 เท่าของขนาดสัญญาณอินพุตที่ต้องการทำการสุ่มและทฤษฎีในส่วนของการเข้ารหัสแบบพัลส์โคดมอดูเลชัน ซึ่งเป็นการดิจิตัล หรือการแปลงข้อมูลอนาล็อกไปเป็นข้อมูลดิจิทัล การเข้ารหัสแบบพัลส์โคดมอดูเลชันนั้น ใช้หลักการวัดค่าต่างระดับระหว่างจุดหรือการควอนไทซ์ ซึ่งต้องนำเอาวงจรกรองความถี่แบบวงจรความถี่ต่ำผ่านมาช่วย เพื่อลดปัญหาความแตกต่างของสัญญาณที่มีมากเกินไป นอกจากนี้มันยังเอาส่วนของหลักการ Companding ไม่ว่าจะวิธีใดๆทั้งสิ้น อีกทั้งยังมีให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เพื่อนำมาช่วยลดปัญหาการรบกวนของการควอนไทซ์เข้ามาพร้อมด้วย รายละเอียดลักษณะของ ทฤษฎีที่กล่าวอ้างอิงถึงคุณได้จากส่วนของทฤษฎีที่ได้อธิบายไปแล้วในบทก่อนๆ

ในระบบที่มีการสุ่มน้อยกว่า NYQUIST RATE นั้น เราเรียกว่า Undersampling สำหรับสัญญาณที่ถูกสุ่มแบบ Undersampling นั้น จะได้ผลลัพธ์ที่มีความถี่เท่ากับ ผลต่างของความถี่ของอัตราการสุ่ม กับความถี่ของสัญญาณที่ป้อนเข้ามา ซึ่งจะมีความผิดเพี้ยนมาก

ส่วนในระบบที่อัตราการสุ่มที่มากจนเกินไป (Oversampling) นั้นอาจมากเกินไปจนทำให้มีความถี่เปลี่ยนแปลงเนื่องที่ของหน่วยความจำมาก และยังลำบากต่อการข้อมูลที่นำไปทำการวิเคราะห์เปรียบเทียบอีกด้วย

**ทฤษฎีของการจดจำ**

แบ่งออกเป็น 2 ส่วน คือ ส่วนของทฤษฎีการจดจำเสียงพูดและส่วนของทฤษฎีการจดจำ

**ผู้พูด**

**ทฤษฎีการจดจำเสียงพูด**

นำเอาทฤษฎีส่วนของการจดจำหน่วยคำที่เปล่งออกมา เป็นลักษณะของการจดจำหน่วย คำแบบเจาะจงผู้พูด เช่น พยางค์เดียว คำ กลุ่มคำ วลี หรือประโยค แต่ไม่ได้จดจำในรูปแบบของ รายละเอียดหน่วยเสียง เช่น สระ พยัญชนะ วรรณยุกต์ เป็นต้น โดยโครงการนี้นำเอาทฤษฎีการจด จำหน่วยคำที่เปล่งออกมาแบบคำเดียว โดยนำเอาหน่วยคำแบบคำเดียวที่โครงการจดจำ นำไปแยก แยะรายละเอียดที่ต้องการ ได้แก่ ขนาดของคำ ค่าของข้อมูลที่ถูกนำไปผ่านการแปลงฟาสฟูเรียร์ เช่นค่าเพาเวอร์สเปกตรัมของข้อมูล ณ ที่ฮาร์โมนิคตำแหน่งต่างๆ เป็นต้น โดยนำเอาข้อมูล ตัวอย่างไปเก็บไว้หลายๆตัว เพื่อหาค่าเฉลี่ยความถี่ของหลักการทางสถิติในที่นี้ใช้ค่าเฉลี่ย และค่าเบี่ยงเบนมาตรฐานที่มีสูตรคั่งข้างล่างแล้วทำการบันทึกค่าไว้เพื่อใช้ในการเปรียบเทียบต่อไป ทั้งนี้เป็นการเอาค่าของเสียงของคนๆเดียวกัน (เจาะจงผู้พูด) มาทำการบันทึกเท่านั้น ในส่วน ของการสังเคราะห์เสียง ก็นำเอาทฤษฎีของการเข้ารหัสพัลส์โค้ดมอดูเลชัน (PCM) แบบ Editing of prerecorded Units เป็นการเอาข้อมูลหน่วยคำที่บันทึกนำมาใช้ในการเปรียบเทียบ ซึ่งใช้ในการ แปลงแบบ Word-waveform coding หรือจากการแปลงรูปคลื่นของคำ ทำให้ระบบการสังเคราะห์ มีลักษณะที่ไม่ซับซ้อนจนเกินไปนัก แต่คุณภาพของเสียงที่ถูกสังเคราะห์จัดอยู่ในเกณฑ์ดีพอใช้

สำหรับการจดจำผู้พูด นำเอาหลักการพิสูจน์ทางพีชคณิตแบบบูลีน ว่าใช่หรือไม่ใช่ผู้พูด โดยการหาค่าความน่าจะเป็น หรือความเป็นไปได้ จะเป็นคำผู้พูดคนที่บันทึกไว้ (แต่ไม่ใช่เป็นการหาว่าจะเป็นคนไหนจากข้อมูลที่จดจำไว้หลายๆคน) โดยนำเอาทฤษฎีส่วนนี้มาร่วมในการ พิจารณาด้วย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### ฟาสต์ฟูเรียร์

ในงานวิจัยนี้ได้ใช้ FFT สำหรับการแปลงสัญญาณเสียงให้ได้เป็นฮาร์โมนิกของเสียง สำหรับนำไปวิเคราะห์เสียงต่อไป

ขั้นตอนวิธีหรือลำดับการในการคำนวณ DFT ให้เร็วที่มีชื่อเรียกว่าการแปลงฟาสต์ฟูเรียร์ หรือ FFT ที่ได้มีการพัฒนาวิธีการต่างๆ กันอย่างมากนั้นเป็นผลมาจากผลงานของ J.W. Cooley กับ J.W. Tukey ที่เสนอไว้ในปี ค.ศ. 1965

FFT จะทำให้การคำนวณ DFT ใช้ การคูณจำนวนเชิงซ้อนเพียง  $N \log_2 N$  ครั้งเท่านั้น หรือจำนวนครั้งในการคูณตัวเลขลดลงไปถึง  $N/(\log_2 N)$  เท่า ตัวอย่างเช่น สำหรับลำดับสัญญาณขนาด  $N = 1024$  จุด การคูณจำนวนเชิงซ้อนลดลงไปได้ถึง 102 เท่า ผลคืออีกประการหนึ่งก็คือ ทำให้การสร้างวงจรเฉพาะเพื่อการคำนวณ DFT ทำได้ง่ายและคำนวณได้เร็วขึ้นเป็นผลให้การประมวลสัญญาณเชิงเลขสามารถนำไปประยุกต์ใช้กับระบบเวลาจริงได้ อย่างไรก็ตามถึงแม้ FFT จะมีชื่อเรียกว่า การแปลงฟาสต์ ฟูเรียร์ แต่ตัว FFT เองนั้นไม่ใช่การแปลงฟูเรียร์เป็นเพียงวิธีการหรือลำดับการในการคำนวณที่ช่วยให้การคำนวณ DFT ซึ่งเป็นการแปลงฟูเรียร์ได้รวดเร็วขึ้น

จนถึงปัจจุบันนี้ได้มีการคิดค้นคิดแปลง และเสนอผลงานเกี่ยวกับ FFT มากมายหลายแบบ โดยทั่วไปแล้วเราอาจแบ่งออกได้เป็นชนิดใหญ่ ๆ 2 ชนิดด้วยกันคือ ชนิดลดทอนทางเวลา (Decimation in Time) กับชนิดลดทอนทางความถี่ (Decimation in Frequency) แต่สำหรับงานวิจัยนี้จะใช้วิธีการลดทอนทางเวลาเพียงอย่างเดียว

การแปลงฟูเรียร์ที่มีลำดับ  $x(m)$  ที่ยาว  $N$  จุด มีสมการดังนี้

$$X(k) = \sum_{m=0}^{N-1} x(m) W^k \quad (6.1)$$

#### ขั้นตอนการลดทอนทางเวลา

เป็นการจัดแบ่งกลุ่มสัญญาณในโดเมนเวลา  $x(m)$  ที่มีขนาด  $N$  จุด ออกเป็นลำดับสัญญาณคู่และคี่ โดยสัญญาณคู่เกิดจากการเอาสัญญาณตำแหน่งเลขคู่มาเรียงกันส่วนที่เหลือก็เป็นสัญญาณคี่ และเราสามารถแบ่งลำดับสัญญาณคู่และคี่ ออกเป็นย่อยๆ ได้อีกจนกระทั่งเหลือการคำนวณ DFT ขนาด 2 จุด ซึ่งเรียกได้ว่าเป็นการแปลงฟาสต์ฟูเรียร์แบบลดทอนทางเวลาซึ่งในงานวิจัยนี้ได้ใช้  $N$  ขนาด 2048 จุด ถ้าเรานิยามให้  $x_E(m)$  เป็นลำดับคู่ และ  $x_O(m)$  เป็นลำดับคี่ เพราะฉะนั้น

$$x_E(m) = x(2m) \quad ; \quad m = 0, 1, \dots, (N/2)-1 \quad (6.2)$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
 $x_O(m) = x(2m + 1) \quad ; \quad m = 0, 1, \dots, (N/2)-1$   
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ด้วยการจัดแบ่งเช่นนี้ ถ้าให้ค่า  $W_N$  แทนค่า  $W$  ของลำดับยาว  $N$  จุด ทำให้การคำนวณ DFT ของลำดับสัญญาณ  $x(m)$  ที่ยาว  $N$  จุดเขียนใหม่ได้เป็น

$$\begin{aligned} X(k) &= \sum_{m=0}^{N-1} x_E(m) (W_N)^{km} + \sum_{m=0}^{N-1} x_O(m) (W_N)^{km} \\ &= \sum_{m=0}^{(N/2)-1} x(2m) (W_N)^{2mk} + \sum_{m=0}^{(N/2)-1} x(2m+1) (W_N)^{(2m+1)k} \end{aligned} \quad (6.3)$$

โดยที่เขียนให้พจน์  $(W_N)^2$  เป็น

$$(W_N)^2 = [\exp(j2\pi/N)]^2 = W_{N/2}$$

ซึ่ง  $W_{N/2}$  หรือค่า  $W$  ของลำดับยาว  $N/2$  จุด เพราะฉะนั้น (6.3) จัดพจน์ใหม่ได้คือ

$$X(k) = \sum_{m=0}^{(N/2)-1} x_E(m) (W_{N/2})^{km} + (W_N)^k \sum_{m=0}^{(N/2)-1} x_O(m) (W_{N/2})^{km}$$

$$X(k) = X_1(k) + (W_N)^k X_2(k) \quad (6.4)$$

โดยที่  $X_1(k)$  และ  $X_2(k)$  แทนผลการแปลง DFT ขนาด  $N/2$  จุดของลำดับ  $x_E(m)$  และ  $x_O(m)$  ตามลำดับ สมการที่ (6.4) แสดงให้เห็นว่าการคำนวณ DFT ขนาด  $N$  จุดนั้นสามารถแบ่งคำนวณย่อยออกเป็นการคำนวณ DFT ขนาด  $N/2$  จุดสองอันได้ และข้อสำคัญคือการคูณจำนวนเชิงซ้อนจะลดลงเหลือ  $2(N/2)^2 = N^2/2$  ครั้ง ซึ่งจะเห็นว่าการลดเวลาการคำนวณลงไปได้ถึง 50 เปอร์เซ็นต์ โดยอาศัยหลักการเดียวกันถ้าเราแบ่งทอนลำดับ  $x_E(m)$  และ  $x_O(m)$  ออกเป็นลำดับคู่และลำดับคี่ลงไปอีกตามลำดับ จนในที่สุดเหลือเป็นลำดับขนาดเพียงสองจุดหรือกล่าวได้ว่า การคำนวณการแปลง DFT ขนาด  $N$  จุดทำได้โดยการคำนวณการแปลง DFT ขนาด 2 จุด จำนวน  $N/2$  ภาคด้วยกัน ข้อสังเกตที่สำคัญก็คือการซอยเพื่อแบ่งลำดับ  $x(m)$  ออกเป็น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ทีละครั้งจนเหลือการคำนวณ DFT ขนาด 2 จุดนี้ สำหรับสัญญาณขนาด  $N$  ลำดับ จะทำการแบ่งออกได้  $\log_2 N$  ครั้ง

ปัญหาต่อมาเมื่อแบ่งย่อยลงไปแล้วจะเอาการแปลง DFT ขนาด 2 จุดจำนวน  $N/2$  นี้มาประกอบกัน หรือรวมตัวกันเพื่อให้เป็นการคำนวณ DFT ขนาด  $N$  ได้อย่างไร เพราะถ้าหากนำมาประกอบกันโดยไม่มีหลักเกณฑ์ ผลการคำนวณ DFT ที่ได้มีค่าผิดไป การนำมาประกอบกันสำหรับลำดับสัญญาณ  $N$  ลำดับอย่างถูกต้อง เราต้องทำการนิยามค่าของสมการ (6.4) สำหรับค่า  $k > N/2$  ด้วย

$$\begin{aligned} X(k) &= X_1(k) + (W_N)^k X_2(k) \quad ; 0 \leq k \leq (N/2)-1 \\ &= X_1(k-N/2) + (W_N)^k X_2(k-N/2) \quad ; N/2 \leq k \leq N-1 \end{aligned} \quad (6.5)$$

พจน์  $(W_N)^k$  ในสมการ (6.5) นี้มีชื่อเรียกว่า **ตัวประกอบหมุน** พจน์นี้เป็นส่วนสำคัญร่วมกับ DFT ขนาดสองจุด หรือ DFT ขนาด  $N/2$  จุด ที่จะนำประกอบกันเข้าเป็น DFT ขนาด  $N$  จุดเหมือนเดิม

ในสมการที่ (6.5) นี้ ถ้าหากเรานำเอาความสัมพันธ์  $(W_N)^{k-N/2} = -(W_N)^k$  แทนลงไป ในสมการนี้เขียนใหม่เป็น

$$\begin{aligned} X(k) &= X_1(k) + (W_N)^k X_2(k) \quad ; 0 \leq k \leq (N/2) \quad (6.6) \\ &= X_1(k-N/2) + (W_N)^{k-(N/2)} X_2(k-N/2) \quad ; N/2 \leq k \leq (N/2)-1 \end{aligned} \quad (6.7)$$

สมการที่ได้นี้เป็นหลักการสำคัญสำหรับ FFT แบบนี้ จากสมการบอกเราว่าในการคำนวณ DFT ของลำดับคู่หนึ่ง จะประกอบด้วยลำดับ

$X(k)$  ในสมการ (6.6) และลำดับ  $X(k)$  ใน (6.7) ซึ่งห่างออกไปจากลำดับ  $X(k)$  ใน (6.6) ไป  $N/2$  จุด สามารถคำนวณได้โดยการใช้การคูณจำนวนเชิงซ้อนเพียงครั้งเดียวเท่านั้น

### ค่าความเบี่ยงเบนมาตรฐาน

การสื่อสารกระจายของข้อมูลโดยใช้ส่วนเบี่ยงเบนมาตรฐานเป็นวิธีที่นักสถิติยอมรับว่าเป็นวิธีที่ใช้วัดการกระจายได้ดีที่สุดเมื่อเทียบกับวิธีการกระจาย 3 วิธี คือ พิธัส ส่วนเบี่ยงเบนมาตรฐานควอไทล์ และส่วนเบี่ยงเบนเฉลี่ย

ใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ทั้งนี้เนื่องจากการวัดการกระจายโดยวิธีนี้ใช้ข้อมูลทุกค่า หรือมีตัวแทนของข้อมูลทุกค่า มาคำนวณ และขจัดปัญหาในการที่ต้องใช้ค่าสัมบูรณ์ให้หมดไปการวัดการกระจายโดยวิธีนี้ออกจากจะให้ค่าการกระจายที่มีความละเอียดถูกต้องและเชื่อถือได้ที่สุด

ในการหาส่วนเบี่ยงเบนเฉลี่ยต้องคิดค่าสัมบูรณ์ของแต่ละค่าของข้อมูลที่ห่างไปจากค่าเฉลี่ยเลขคณิตเพราะไม่เช่นนั้นแล้วส่วนเบี่ยงเบนจะเป็นศูนย์ การนำค่าส่วนเบี่ยงเบนของข้อมูลทั้งหมดที่ห่างไปจากค่าเฉลี่ยเลขคณิตมายกกำลังสองแล้วนำมาบวกกันและหารด้วยจำนวนข้อมูลทั้งหมดแล้วถอดรากที่สอง ค่าที่ได้นี้เรียกว่า ส่วนเบี่ยงเบนมาตรฐาน ถ้า  $X_1, X_2, X_3, \dots, X_N$  เป็นค่าของข้อมูล  $N$  จำนวน และค่าเฉลี่ยของข้อมูลชุดนี้คือ  $\bar{X}$

$$\text{ส่วนเบี่ยงเบนมาตรฐาน}(s) = \sqrt{\sum_{i=1}^N (X_i - \bar{X})^2 / N}$$

ค่ามาตรฐาน คือค่าตัวเลขที่ใช้เปรียบเทียบค่าของข้อมูลของตัวแปรตั้งแต่สองตัวขึ้นไปว่า ข้อมูลตัวใดดีกว่ากัน โดยนำค่าจริงที่ได้ลบด้วยค่าเฉลี่ยแล้วหารด้วยส่วนเบี่ยงเบนมาตรฐาน

ถ้าให้  $Z_i$  เป็นค่ามาตรฐานของข้อมูล

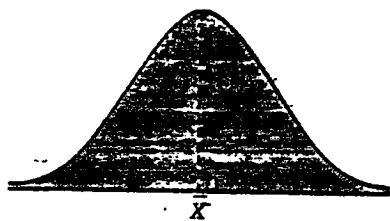
$X_i$  เป็นค่าของข้อมูลที่มีค่าเฉลี่ยและส่วนเบี่ยงเบนมาตรฐานเป็น  $\bar{X}$  และ  $s$  ตามลำดับ

$$Z_i = (X_i - \bar{X}) / s$$

การแจกแจงปกติ เป็นการแจกแจงความถี่ของข้อมูลซึ่งให้เส้นโค้งมีลักษณะเป็นรูประฆัง

คุณสมบัตินี้ของเส้นโค้งปกติ

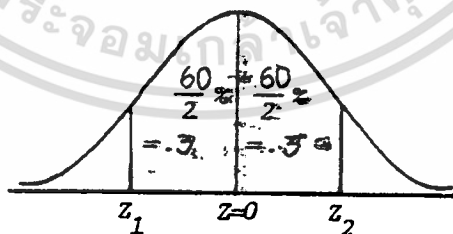
1. ค่าเฉลี่ยเลขคณิต มัชฐาน และฐานนิยม จะมีค่าเท่ากันและจะอยู่ ณ จุดที่เส้นตรงที่ลากผ่านจุดสูงสุดของเส้นโค้งนั้นตั้งฉากกับแกนนอน
2. เส้นโค้งจะมีเส้นตรงตั้งฉากที่ลากผ่านค่าเฉลี่ยเลขคณิตเป็นแกนสมมาตร
3. เส้นโค้งจะเข้าใกล้แกนระนาบ เมื่อเราค่อยปลายเส้นโค้งทั้งสองข้างให้ห่างจากค่าเฉลี่ยเลขคณิตออกไปแต่ให้ตัดแกนนอน
4. พื้นที่ใต้เส้นโค้งปกติมีค่าเท่ากับ 1 เสมอ



รูปที่ 6.3 แสดงเส้นโค้งปกติรูประฆังคว่ำ

สมการของเส้นโค้งปกติขึ้นอยู่กับค่าเฉลี่ยเลขคณิตและค่าส่วนเบี่ยงเบนมาตรฐานเราสามารถหาพื้นที่ใต้เส้นโค้งปกติได้โดยใช้ตารางแสดงพื้นที่ใต้โค้ง โดยการเปลี่ยนค่าจากการสังเกต  $X$  ให้เป็นค่ามาตรฐาน  $Z$  โดยใช้สูตร  $Z = (X - \bar{X}) / s$  ซึ่งถ้าค่าจากการสังเกต  $X$  มีการแจกแจงปกติ โดยมีค่าเฉลี่ยเลขคณิตเป็น  $\bar{X}$  และส่วนเบี่ยงเบนมาตรฐานเป็น  $s$

ตามหลักการทางสถิติ ข้อมูลที่จะมีความถูกต้องและใกล้เคียงกับฐานข้อมูลจะเป็นมีค่าใกล้เคียงกับค่าเฉลี่ยทางสถิติโดยค่าของข้อมูลต้องมีค่าอยู่ระหว่าง บวกลบค่าเบี่ยงเบนมาตรฐานกับค่าเฉลี่ยทางสถิติ



รูปที่ 6.4 แสดงพื้นที่ใต้โค้งที่ใช้ในการวิเคราะห์

จากรูป ข้อมูลของเสียงที่นำมาเปรียบเทียบจะเป็นเสียงตามฐานเสียงที่เก็บไว้หรือไม่หาได้จากการ นำเสียงที่ต้องการทราบมาคำนวณหาจุดตกของค่าในกราฟระฆังคว่ำ ถ้าค่าที่ได้ตกเอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

อยู่ในพื้นที่ที่กำหนด ( พื้นที่ใต้กราฟที่กำหนดมีค่า 60 เปอร์เซ็นต์ โดยมีค่า  $Z_1 = 1$  และ  $Z_2 = 2$ ) จะถือว่าค่าของเสียงพูดนั้นเป็นค่าที่ยอมรับได้

แต่ถ้าค่าของเสียง ไม่ตกอยู่ในช่วงที่กำหนดก็จะไม่นำค่าของเสียงนั้นมาคำนวณหา ความถูกต้องของเสียง

ในการหาความถูกต้องของเสียงจะนำค่าฮาร์โมนิกของเสียงที่ต้องการเปรียบเทียบมา เปรียบเทียบกับฮาร์โมนิกของฐานเสียงทุกตัว โดยการเปรียบเทียบจะใช้วิธีการเปรียบเทียบทาง สถิติที่กล่าวมาแล้ว หลังจากเปรียบเทียบฮาร์โมนิกทุกตัวแล้วจะนำฮาร์โมนิกที่ยอมรับว่าถูกต้อง มาคำนวณหาความถูกต้องของเสียงอีกครั้งหนึ่ง คือ ค่าของฮาร์โมนิกที่ถูกต้องใกล้เคียงกับเสียง โคที่สุก โดยคำนวณออกมาเป็น เปอร์เซ็นต์ความถูกต้องของเสียง

### สเปกตรัมของเสียง

ฮาร์โมนิกของเสียงที่ทำการเก็บ วิเคราะห์ เปรียบเทียบจะเป็นฮาร์โมนิกซึ่งได้จากการนำ ค่าของส่วนจริงและค่าของส่วนจินตภาพมาถ่วงแล้วนำมาบวกกันก่อนนำมาออครากที่ สอง

$$P = \sqrt{\text{Re}^2 + \text{Im}^2}$$

P พาวเวอร์สเปกตรัมของเสียง

Re ค่าของส่วนจริงซึ่งได้จากการแปลง FFT

Im ค่าของส่วนจินตภาพซึ่งได้จากการแปลง FFT

## บทที่ 7

### การทดลองและผลการทดลอง

#### 7.1 การทดลอง

ในโครงการนี้ การทดลองจะใช้ตัวอย่างของเสียงทั้งหมด 4 คำ โดยจะแบ่งลักษณะคำออกเป็นสองกลุ่มคือ

1. คำที่มีหนึ่งพยางค์ มีสองคำคือ
  - DIR เป็นพยางค์เดียวที่เป็นเสียงยาว
  - ONE เป็นพยางค์เดียวที่เป็นเสียงสั้น
2. กลุ่มคำที่มีสองพยางค์
  - SCAN พยางค์แรกจะเป็นเสียงสั้น ส่วนพยางค์ที่สองจะเป็นเสียงยาว
  - CHK พยางค์แรก และพยางค์ที่สองจะเป็นเสียงสั้น

สำหรับการทดลองนี้จะใช้ผู้พูดสองคน โดยแต่ละคนจะพูด 20 ครั้งในแต่ละคำ เพื่อนำไปวิเคราะห์และเก็บเป็นฐานเสียงเพื่อนำมาเปรียบเทียบ

#### 7.2 วิธีการทดลอง

1. ให้นักทดลองพูดคำสั่งซึ่งมีอยู่ 4 คำ ได้แก่
  - DIR โดยออกเสียงว่า เคอร์
  - ONE โดยออกเสียงว่า หนึ่ง
  - CHK โดยออกเสียงว่า เช็ก-คิต
  - SCAN โดยออกเสียงว่า สะ-แกน

ในการทดลองจะพูดคำสั่งที่กล่าวมาแล้ว คำสั่งละ 10 ครั้ง

2. เมื่อพูดในแต่ละครั้งเมื่อเสร็จสิ้นจะปรากฏเปอร์เซ็นต์ของความถูกต้องของคำสั่งแต่ละคำให้บันทึกเปอร์เซ็นต์ความถูกต้องไว้เพื่อการเปรียบเทียบในแต่ละคำ

3. เมื่อพูดในคำสั่งทุกคำเสร็จสิ้น ทดลองพูดคำที่ไม่มีใน ฐานข้อมูล เช่น

- สอง ออกเสียง สอง ซึ่งมีลักษณะการออกเสียงคล้ายกับ DIR
- MEM ออกเสียง เม้ม ซึ่งมีลักษณะการออกเสียงคล้ายกับ ONE
- EDIT ออกเสียง อี-คิต ซึ่งมีลักษณะการออกเสียงคล้ายกับ SCAN
- VIRUS ออกเสียง ไว-รัส ซึ่งมีลักษณะการออกเสียงสองพยางค์ที่เกือบต่อ

เนื่องกัน

การออกเสียงนอกเหนือคำสั่งในคาต้าเบส ออกเสียงคำละ 2 ครั้ง

เอกสารนี้ 4. บันทึกเปอร์เซ็นต์ความถูกต้องไว้เพื่อการเปรียบเทียบ ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 7.3 ผลการทดลอง



รูปที่ 7.1 a. แสดงรูปคลื่นเสียงที่พูดคำว่า DIR ก่อนที่จะทำการแปลง FFT  
 เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ไว้สำหรับใช้ในงานวิจัยเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
 b. แสดงฮาร์โมนิกของเสียงที่ได้จากการแปลง FFT  
 ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อพูดออกเสียงคำว่า DIR

เมื่อทำการวิเคราะห์และเปรียบเทียบสัญญาณเสียงที่พูดเข้ากับคำค่าเบสแล้ว  
จะปรากฏข้อมูล

Word dir is correct 78.46 %

Word one is correct 20.00 %

Word ohk is correct 3.07 %

Word sean is correct 4.61 %

Word command is dir

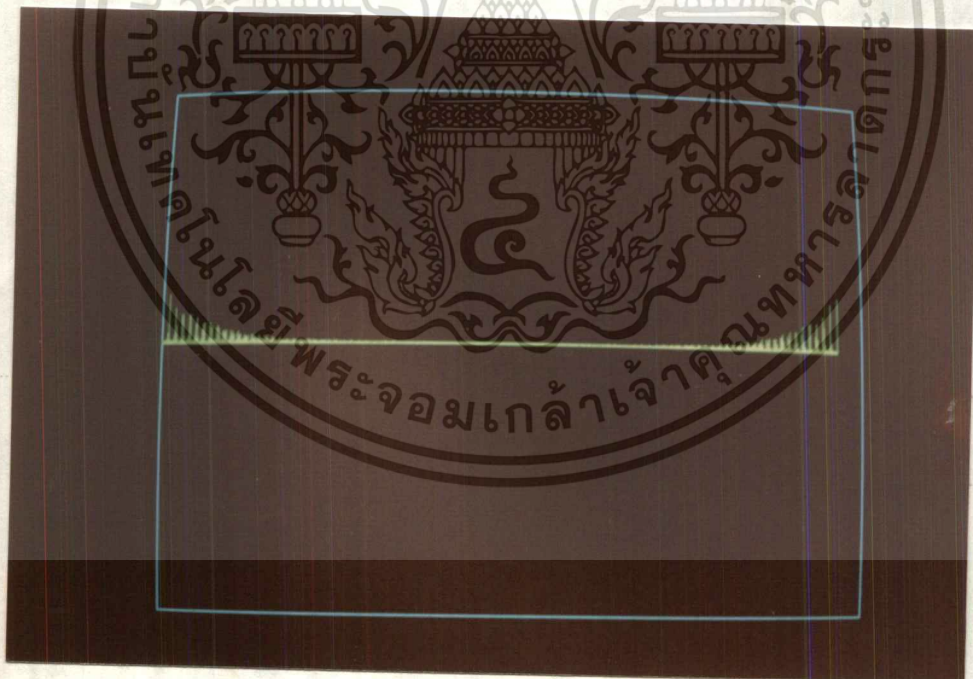
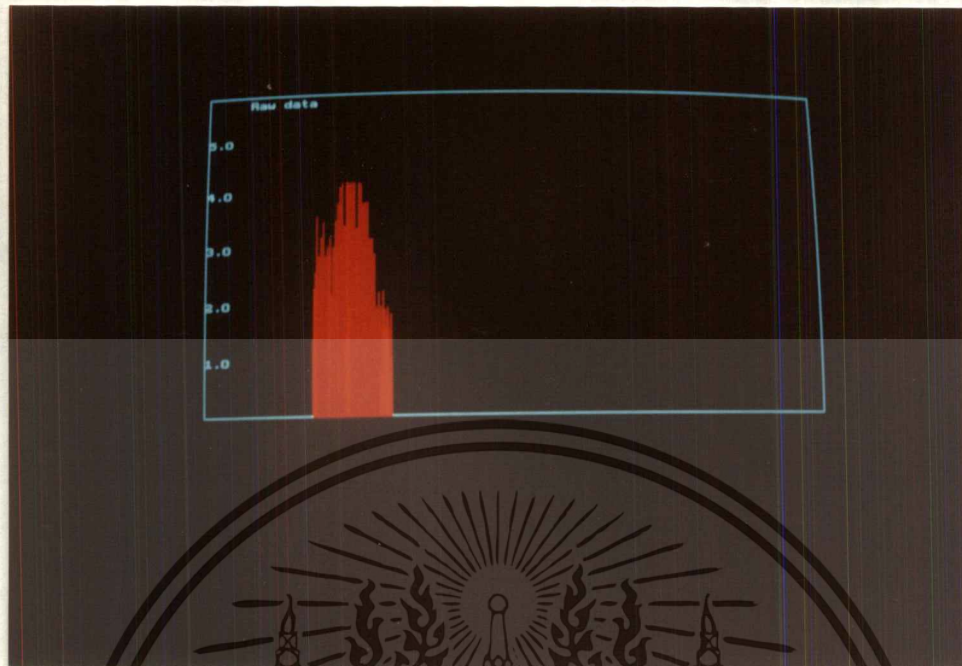
แล้วทำการบันทึกผล 10 ครั้งตามตาราง

ครั้งที่	DIR	ONE	CHK	SCAN
1	78.46	20.00	3.07	4.61
2	80.58	20.00	1.53	3.07
3	83.07	18.46	3.07	6.15
4	60.00	12.30	3.07	6.15
5	83.84	20.00	4.61	4.61
6	81.53	21.07	3.07	6.15
7	69.23	20.00	3.07	7.69
8	49.48	18.46	1.53	7.69
9	92.30	15.38	1.53	4.61
10	67.69	30.84	1.53	4.61

ตารางที่ 7.1 แสดงเปอร์เซ็นต์ของการพูดคำว่า DIR 10 ครั้ง

เมื่อพูด DIR 10 ครั้ง จะเห็นว่าเปอร์เซ็นต์ของ DIR จะสูงกว่าเปอร์เซ็นต์ของคำอื่น  
ทั้ง 10 ครั้ง คิดเป็นเปอร์เซ็นต์เฉลี่ยได้ 74.62 เปอร์เซ็นต์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



(b)

รูปที่ 7.2 a. แสดงรูปคลื่นเสียงที่พูดคำว่า ONE ก่อนที่จะทำการแปลง FFT

b. แสดงฮารโมนิกของเสียงที่ได้จากการแปลง FFT

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
 เมื่อพูดออกเสียงคำว่า ONE  
 ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อทำการวิเคราะห์และเปรียบเทียบสัญญาณเสียงที่พูดเข้ากับคำค่าเบสแล้ว  
จะปรากฏข้อมูล

Word dir is correct 26.15 %

Word one is correct 83.84 %

Word ohk is correct 1.53 %

Word scan is correct 3.07 %

Word oommand is one

แล้วทำการบันทึกผล 10 ครั้งตามตาราง

ครั้งที่	DIR	ONE	CHK	SCAN
1	26.15	83.84	1.53	3.07
2	18.46	81.53	2.00	3.17
3	24.61	61.53	1.53	4.61
4	58.46	26.15	2.07	7.69
5	21.53	80.00	2.61	1.53
6	24.65	82.30	3.58	3.23
7	49.53	36.76	2.92	3.69
8	26.38	67.69	1.58	4.53
9	24.85	79.92	1.15	3.27
10	26.87	85.38	1.62	3.92

ตารางที่ 7.2 แสดงเปอร์เซ็นต์ของการพูดคำว่า ONE 10 ครั้ง

เมื่อพูด ONE 10 ครั้ง จะเห็นว่าเปอร์เซ็นต์ของ ONE จะสูงกว่าเปอร์เซ็นต์ของคำอื่น  
8 ครั้ง คิดเป็นเปอร์เซ็นต์เฉลี่ยได้ 68.51 เปอร์เซ็นต์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 7.3 a. แสดงรูปคลื่นเสียงที่พูดคำว่า CHK ก่อนที่จะทำการแปลง FFT

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับงานวิจัยของศูนย์วิจัยและพัฒนาเทคโนโลยีการแปลงสัญญาณเสียงพูดเพื่อใช้ในการศึกษาวิจัยเท่านั้น ไม่ควรเอาไปใช้ประโยชน์ด้านการค้า

b. แสดงฮาร์โมนิกของเสียงที่ได้จากการแปลง FFT

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อพูดออกเสียงคำว่า CHK

เมื่อทำการวิเคราะห์และเปรียบเทียบสัญญาณเสียงที่พูดเข้ากับคำค่าเบสแล้ว  
จะปรากฏข้อมูล

Word dir is correct 2.65 %

Word one is correct 3.34 %

Word chk is correct 83.53 %

Word scan is correct 23.07 %

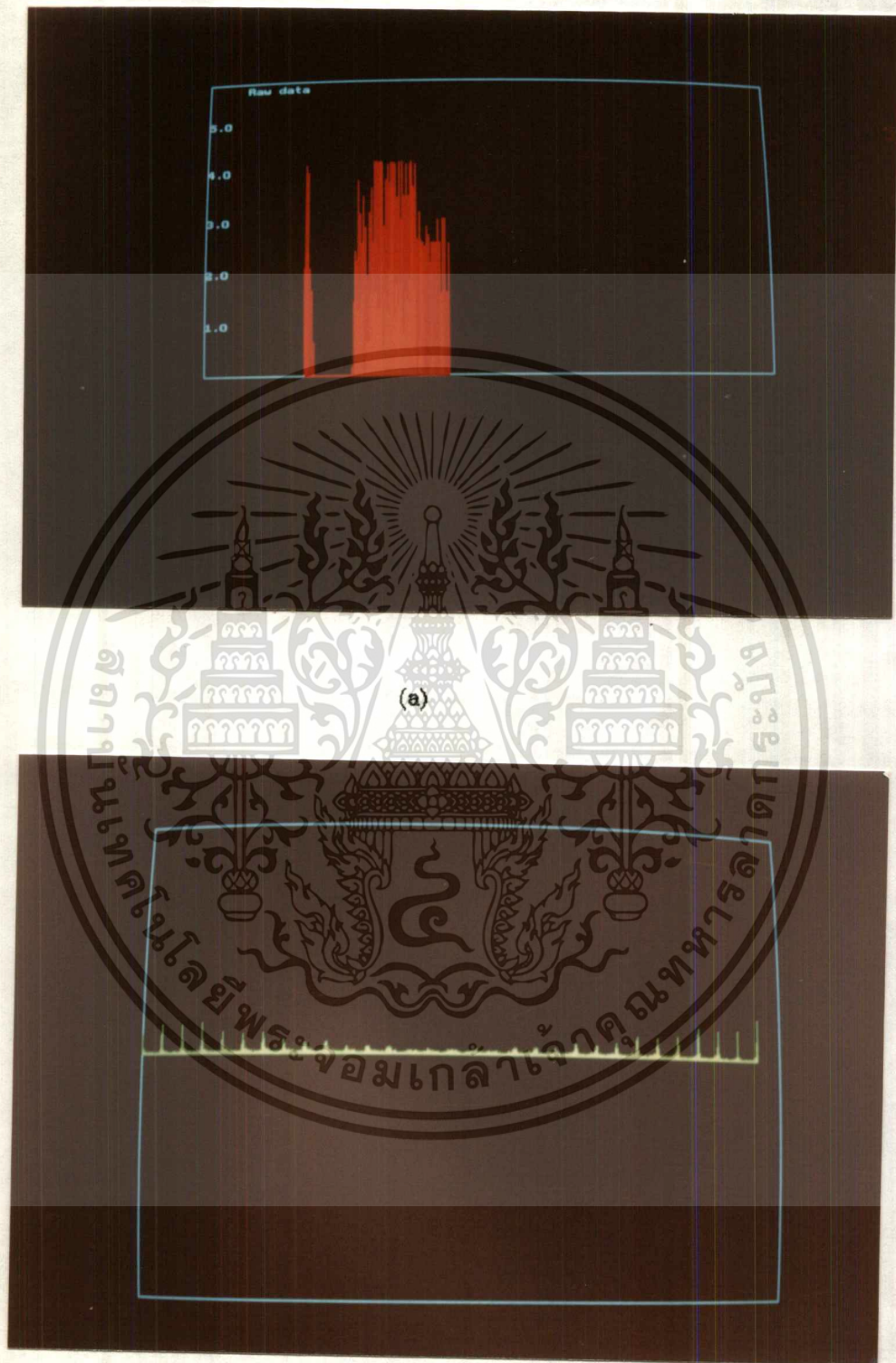
Word command is ohk

แล้วทำการบันทึกผล 10 ครั้งตามตาราง

ครั้งที่	DIR	ONE	CHK	SCAN
1	2.65	1.38	80.23	17.69
2	3.92	1.15	83.84	24.61
3	3.68	2.76	82.30	23.84
4	3.76	2.30	63.07	29.23
5	1.23	3.77	66.15	67.38
6	2.69	3.84	86.15	67.69
7	6.70	2.53	81.53	26.15
8	5.38	1.58	54.51	59.37
9	7.76	5.38	89.23	27.69
10	3.23	1.61	83.84	26.87

ตารางที่ 7.3 แสดงเปอร์เซ็นต์ของการพูดคำว่า CHK 10 ครั้ง

เมื่อพูด CHK 10 ครั้ง จะเห็นว่าเปอร์เซ็นต์ของ CHK จะสูงกว่าเปอร์เซ็นต์ของคำอื่น  
8 ครั้ง คิดเป็นเปอร์เซ็นต์เฉลี่ยได้ 77.08 เปอร์เซ็นต์



(b)

รูปที่ 7.4 a. แสดงรูปคลื่นเสียงที่พูดคำว่า SCAN ก่อนที่จะทำการแปลง FFT

เอกสารนี้เป็นเอกสารที่สแกนไว้สำหรับงานวิจัยส่วนตัวเท่านั้น ไม่ลงชื่อหน้าไปใช้ประโยชน์ด้านการค้า

b. แสดงฮาร์โมนิกของเสียงที่ได้จากการแปลง FFT

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อพูดออกเสียงคำว่า SCAN

เมื่อทำการวิเคราะห์และเปรียบเทียบสัญญาณเสียงที่พูดเข้ากับคำค่าเบสแล้ว

จะปรากฏข้อมูล

Word dir is correct 4.35 %

Word one is correct 2.43 %

Word chk is correct 24.35 %

Word scan is correct 85.23 %

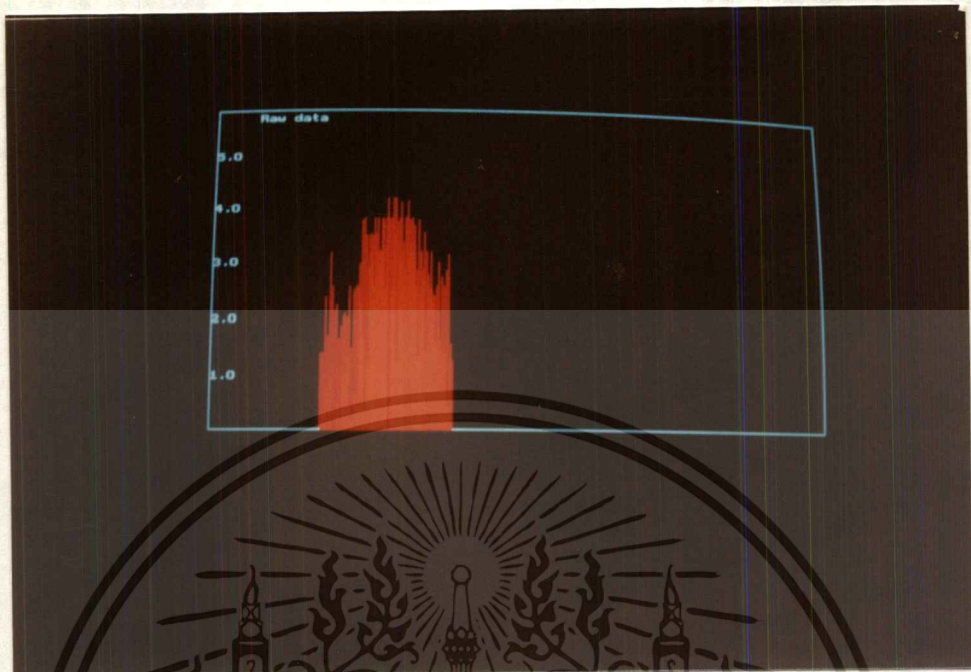
Word command is scan

แล้วทำการบันทึกผล 10 ครั้งตามตาราง

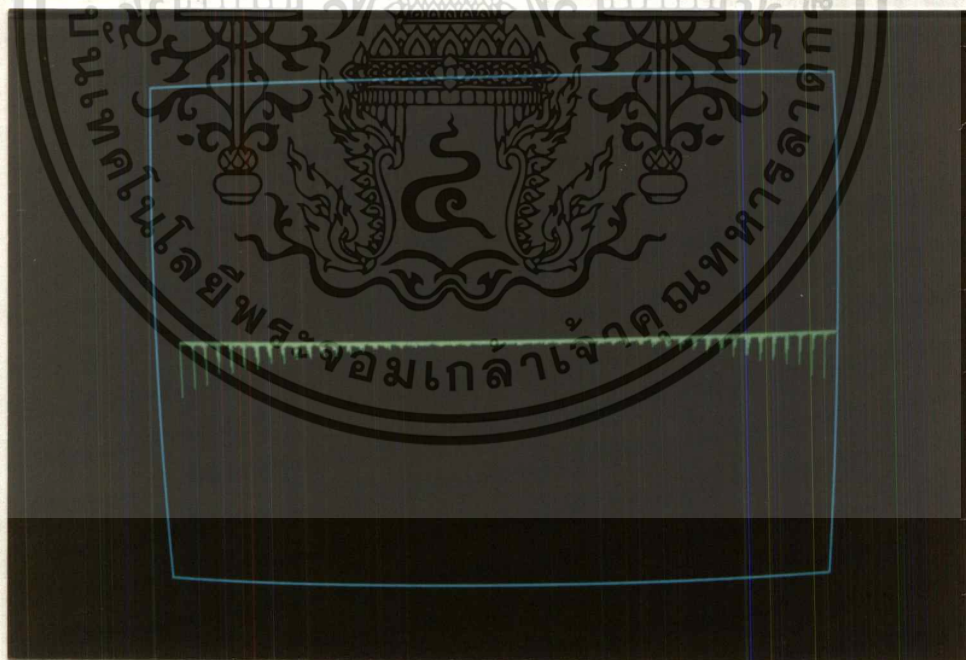
ครั้งที่	DIR	ONE	CHK	SCAN
1	13.84	20.00	47.69	86.92
2	7.38	11.53	30.76	89.23
3	15.34	8.64	32.30	60.00
4	10.76	12.70	29.23	55.38
5	9.23	6.29	49.23	80.53
6	7.08	15.38	61.69	59.84
7	8.73	8.09	35.38	95.38
8	10.96	7.34	53.84	83.07
9	6.23	12.16	55.38	54.61
10	13.32	13.36	35.46	85.34

ตารางที่ 7.4 แสดงเปอร์เซ็นต์ของการพูดคำว่า SCAN 10 ครั้ง

เมื่อพูด SCAN 10 ครั้ง จะเห็นว่าเปอร์เซ็นต์ของ SCAN จะสูงกว่าเปอร์เซ็นต์ของคำอื่น 8 ครั้ง คิดเป็นเปอร์เซ็นต์เฉลี่ยได้ 73.90 เปอร์เซ็นต์



(a)



(b)

รูปที่ 7.5 a. แสดงรูปคลื่นเสียงที่พูดคำที่ไม่มีในคาด้าเบสก่อนที่จะทำการแปลง FFT

เอกสารนี้เป็นเอกสารที่แสดงฮาร์โมนิกของเสียงที่ได้จากการแปลง FFT นุญญาติให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อออกเสียงที่ไม่มีข้อมูลในคาคำเบส ซึ่งในการทดลองจะยกตัวอย่าง 4 คำคือ สอง  
EDIT VIRUS MEM

เมื่อทำการวิเคราะห์และเปรียบเทียบสัญญาณเสียงที่พูดเข้ากับคาคำเบสแล้ว  
จะปรากฏข้อมูล ตัวอย่างเป็นการ คำว่า สอง

Word dir is correct 41.53 %

Word one is correct 24.61 %

Word ohk is correct 3.06 %

Word scan is correct 10.76 %

Word command is dir

แล้วทำการบันทึกผล 10 ครั้งตามตาราง

	DIR	ONE	CHK	SCAN
สอง	41.53	24.61	3.06	10.76
สอง	47.69	21.58	1.58	13.84
EDIT	10.61	12.20	49.23	27.69
EDIT	13.27	16.92	43.07	37.69
VIRUS	37.69	35.18	40.61	36.26
VIRUS	36.38	30.67	41.46	44.46
MEM	20.71	27.69	1.70	3.37
MEM	16.21	25.39	3.75	3.96

ตารางที่ 7.5 แสดงเปอร์เซ็นต์ของคำที่ไม่มีเก็บในคาคำเบส

จะเห็นว่าเปอร์เซ็นต์ของคำที่มีลักษณะเหมือนคาคำเบสคำใด เปอร์เซ็นต์คาคำเบส  
ของคำนั้นจะสูง

## สรุปผลการทดลอง

ในการทดลองจะให้คนหนึ่งคนพูด 10 ครั้งในแต่ละคำและพูดคำนอกเหนือ data base

1. dir มีเปอร์เซ็นต์ในการวิเคราะห์ถูกต้องประมาณ 74.62 เปอร์เซ็นต์ ซึ่งในการพูด 10 ครั้งจะถูกต้องประมาณ 9 ครั้ง ผิด 1 ครั้ง
2. one มีเปอร์เซ็นต์ในการวิเคราะห์ถูกต้องประมาณ 68.51 เปอร์เซ็นต์ ซึ่งในการพูด 10 ครั้งจะถูกต้องประมาณ 8 ครั้ง ผิด 2 ครั้ง
3. soan มีเปอร์เซ็นต์ในการวิเคราะห์ถูกต้องประมาณ 77.08 เปอร์เซ็นต์ ซึ่งในการพูด 10 ครั้งจะถูกต้องประมาณ 8 ครั้ง ผิด 2 ครั้ง
4. okeokdisk มีเปอร์เซ็นต์ในการวิเคราะห์ถูกต้องประมาณ 73.90 เปอร์เซ็นต์ ซึ่งในการพูด 10 ครั้งจะถูกต้องประมาณ 7 ครั้ง ผิด 3 ครั้ง
5. เสียงที่ไม่มีใน data base จะมีเปอร์เซ็นต์ที่ถูกต้องเฉลี่ยประมาณ 10-20 เปอร์เซ็นต์ใกล้เคียงกันทุกคำ โดยคำผิดพลาดที่พูดเข้าไปใกล้เคียงกับคำใดเปอร์เซ็นต์การวิเคราะห์ของคำนั้นจะสูงกว่าคำอื่นเล็กน้อย

## สรุป

จากการทดลองข้อผิดพลาดที่เกิดขึ้นเนื่องมาจาก

1. การ Sampling ของฮาร์ดแวร์ได้ข้อมูลไม่ละเอียดพอ บางครั้งอาจมีการผิดพลาดได้เนื่องจากข้อมูลบางส่วนไม่ได้ถูกนำมาวิเคราะห์
2. การตัดทอนข้อมูลก่อนการแปลง FFT อาจมีบางส่วนที่หายหรือเพิ่มเข้ามาของข้อมูลที่ไม่ต้องการ
3. การใช้ FFT บางครั้งข้อมูลจะต้องถูกตัดทอนทิ้ง เนื่องจากการวิเคราะห์ของ FFT ต้องใช้ข้อมูลขนาด  $2^N$
4. การวิจัยสามารถนำประยุกต์ใช้งานได้กับการสั่งงานเครื่องจักรที่มีคำสั่งไม่มากนัก เช่น การสั่งงานแขนกล

## แนวทางการแก้ไข

### HARDWARE

1. วงจร A/D ควรมีความเร็วมากกว่าในการทดลองนี้
2. คำนับสัญญาณแอดเครสให้กับ RAM ทำงานไม่สัมพันธ์กันกับ RAM และ RAM ควรมีประสิทธิภาพในการรับแอดเครสที่ดีกว่านี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. ตัวพอร์ต 8255 ซึ่งมีหน้าที่รับสัญญาณ RAM เต็ม จะมีความร้อนสูงในขณะที่ใช้งาน ทำให้เกิดการเสียหายได้ง่าย ควรปรับปรุงวงจรพอร์ตส่วนนี้

#### ข้อสังเกต

สัญญาณที่แสดง RAM เต็ม บ่อยครั้งที่ทำงานบกพร่อง หรือไม่ทำงานเลย ซึ่งอาจมีสาเหตุมาจาก

1. ความร้อนของตัวพอร์ตสูง ทำให้ตัวพอร์ตเสียหายเร็วกว่าปกติ
2. อาจเกิดจากควมับแอคเครตของ RAM เกิดเสีย

#### SOFTWARE

1. ไม่ควรใช้ภาษา C เหมือนในการทดลองนี้ เพราะภาษา C มีตัวแปรจำนวนจำกัด และไม่เหมาะกับการวิเคราะห์เชิงสถิติ
2. ภาษา C มีข้อจำกัดในการเรียนรู้และจดจำข้อมูลเป็นจำนวนมาก และมีการประมวลผลที่ค่อนข้างช้า
3. การใช้วิธีการแปลงฟูเรียร์ไม่ใช่วิธีที่ดีที่สุดในกระบวนการ DIGITAL SIGNAL PROCESSING (DSP) เพราะจะมีข้อผิดพลาดในเรื่องสัญญาณอินพุตที่เป็นโคไซน์ให้แก่การแปลงฟูเรียร์ เพราะในบางครั้งเราไม่สามารถจำกัดจำนวนข้อมูลและอัตราการสุ่มให้คงที่ได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## บทที่ 8 การใช้โปรแกรม

ในการวิจัย ไฟล์โปรแกรมจะอยู่ในชื่อ `Getdata.EXE` ซึ่งการใช้โปรแกรมคำสั่งนี้ต้องมีไฟล์ Data base ประมาณ 70 ไฟล์ ซึ่งแบ่งออกเป็น

1. ไฟล์เก็บข้อมูลจริง
2. ไฟล์เก็บค่าเฉลี่ยข้อมูล
3. ไฟล์เก็บค่า S.D.
4. ไฟล์เก็บค่าเฉลี่ย S.D.

และจะต้องมีไฟล์ `Shortoom.AND` สำหรับเก็บคำสั่งทั้งหมดของเสียง ในการพล็อตกราฟจะต้องใช้ไฟล์ `EGABGI` กับ `BGAVGABGI` ซึ่งไฟล์ทั้งหมดดังที่กล่าวมาจะต้องอยู่ในรูปโคเรกทอรีเดียวกัน เพื่อให้โปรแกรมสามารถเรียกใช้งานได้

การรันโปรแกรม

1. เปิดปุ่ม POWER
2. เปลี่ยนโคเรกทอรีที่มีไฟล์ `Getdata.EXE`
3. เรียกใช้โปรแกรม `Getdata.EXE`
4. และจะปรากฏข้อความต่อไปนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

What do you want to work press choose your choice

1. Write base sound
2. To command
3. To exit

Press your choice : °



เลือก 1. เมื่อต้องการวิเคราะห์และเก็บฐานข้อมูลของเสียง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Does word you want get is old word or new word (Old:New) :



ให้เลือกคำที่จะเก็บว่าเป็นคำใหม่หรือคำที่มีอยู่เดิม

- กศอักษร O เมื่อต้องการเก็บคำที่มีอยู่แล้ว
- กศอักษร N เมื่อต้องการเก็บคำใหม่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

What is word you want get : two



ให้พิมพ์คำสั่งที่ต้องการเก็บข้อมูลเพื่อเป็นฐานเสียง และให้กด Enter เพื่อปรากฏ  
ข้อความต่อไป จากตัวอย่างข้างบนจะเป็นการเก็บคำว่า two

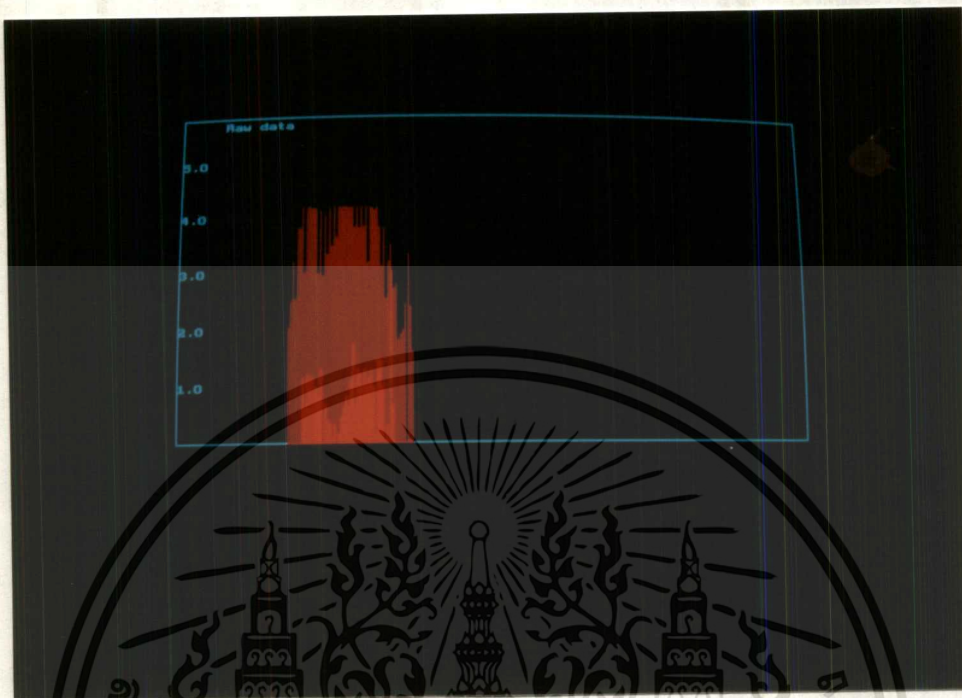
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

What is word you want get : two  
press enter when ready

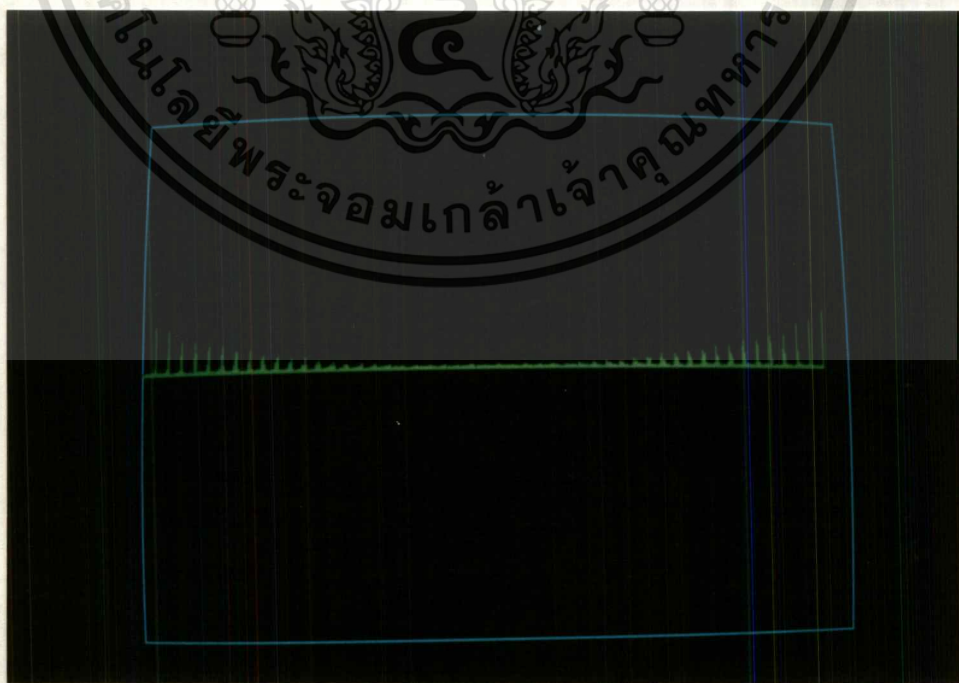


เมื่อพร้อมแล้วให้กด Enter แล้วพูดเสียงของคำนั้นเข้าไมโครโฟนก็จะสามารถเก็บเสียงที่  
ต้องการได้ประมาณสองวินาทีหลังจากกดEnter เมื่อเก็บสัญญาณเสียงแล้วจะปรากฏรูปของ  
สัญญาณเสียง และให้กด Enter อีกครั้งหนึ่งจะปรากฏรูปของฮารโมนิกของสัญญาณที่แปลง FFT  
แล้ว กด Enter กด Enter อีกครั้งจะปรากฏข้อความต่อไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 8.1 แสดงสัญญาณเสียงที่ได้จากการพูดเพื่อนำไปแปลง FFT



รูปที่ 8.2 แสดงฮาร์โมนิกของเสียงที่ได้จากการแปลง FFT

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
 ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

These are data do you want to compile (Yes/No) :

เมื่อปรากฏข้อความนี้จะเป็นการตัดสินใจว่าจะเก็บ data base หรือไม่ เพราะบางครั้งจะมี error เกิดขึ้นถ้าเก็บเข้าไปจะทำให้ชุดของข้อมูลมีความถูกต้องน้อยลง

- ถ้ากด Y หมายถึงเก็บ
- ถ้ากด N หมายถึงไม่เก็บ

แล้วจะปรากฏข้อความต่อไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

These are data do you want to compile (Yes.No) :

Please wait..

Compile

Do you want to exit press 'n'

ถ้าต้องการออกจากโปรแกรมให้กด 'n' ถ้าไม่ต้องการออกจากโปรแกรมให้กดคีย์ใดๆ  
แล้วจะปรากฏข้อความเหมือนตอนเริ่มต้นการรันโปรแกรม

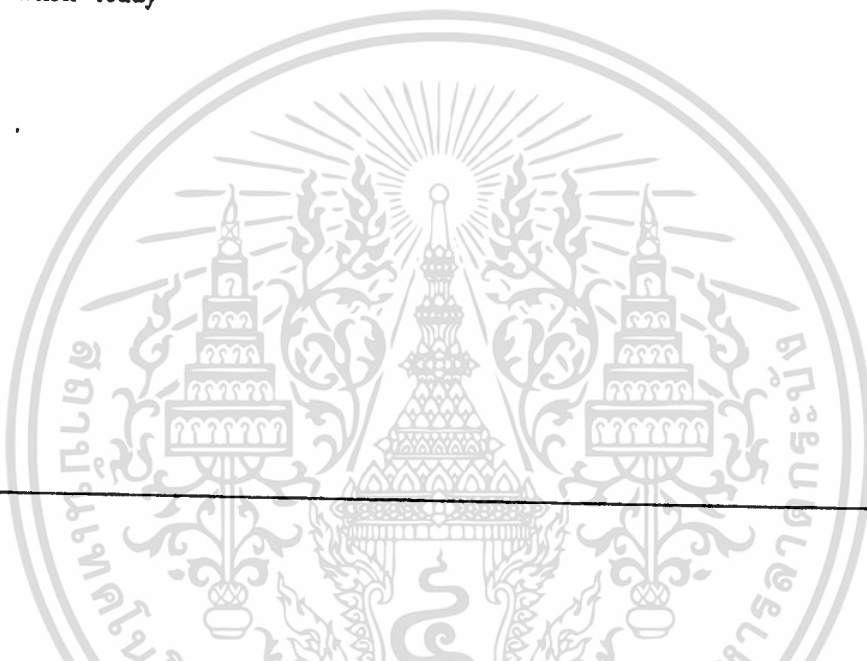
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

What do you want to work press choose your choice

1. Write base sound
2. To command
3. To exit

Press your choice : 2

press enter when ready



เมื่อกด 2 จะปรากฏข้อความ press enter when ready พอกด enter แล้วให้พูดคำสั่งที่ต้องการผ่านไมโครโฟนเข้าไปแล้วจะปรากฏรูปสัญญาณเสียง เมื่อกด enter จะปรากฏสารโมนิกของเสียง และเมื่อกด enter อีกครั้งหนึ่งคอมพิวเตอร์ก็จะทำการเปรียบเทียบสัญญาณเสียงที่เข้าไปกับฐานข้อมูลเสียงที่ได้เก็บไว้แล้วแสดงผลทางจอคอมพิวเตอร์

Word dir is correct \_ %

Word one is correct \_ %

Word scan is correct \_ %

Word checkdisk is correct \_ %

Word command is \_ ซึ่งจะเป็นคำที่มีเปอร์เซ็นต์สูงสุด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 8.3 แสดงสัญญาณเสียงที่ได้จากการพูดเพื่อนำไปแปลง FFT

รูปที่ 8.3 แสดงฮาร์โมนิคของเสียงที่ได้จากการแปลง FFT

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

#include <bios.h>
#include <dos.h>
#include <conio.h>
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include <alloc.h>
#include <math.h>
#include <float.h>
#include <graphics.h>
#include <process.h>
#include <time.h>

#define PI 3.142857143L
#define RA 0.017460317L
#define Max_array 2048
#define porta 0x031c
#define portb 0x031d
#define portc 0x031e
#define pctr1 0x031f
#define SAMPLES      2048
#define FREQ          50
#define AMPLITUDE_MAX 5
#define AMPLITUDE_MIN 0
#define MAX_HARMONIC SAMPLES

typedef struct complex_struct {
    float re;
    float im;

```

เอกสารนี้เป็นเอกสารตัวอย่างสำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    } complex;
```

```
void sampling(complex *square_wave, int samples);
int bit_reverse(complex *input_data, int bit, complex *rv_data);
void fft(complex *input_data, int samples, complex *fft_data);
void dit_fft(complex *input_data, int samples, complex *fft_data);
void cal_amp(complex *fft_data, int samples, float *amp);
void W(int n1, int n2, complex *result);
void complex_add(complex *result, complex adder1, complex adder2);
void complex_subtract(complex *result, complex subtracter1, complex subtracter2);
void complex_multiply(complex *result, complex multiplicand, complex multiplier);
void graphic(void);
void Initialize_Graphics_Mode(void);
void openfile(float *amp);
int filecomp(float *amp,int samples);
savefile();
void grap(float *amp);

int q,maxdif,PRI,END,ii[7];
float R[Max_array];
int data[20000],in_i,c,h;
char word[15],wordcomp[9][15];
```

```
main()
```

```
{
    FILE *in,*in_randow,*in1,*longwordcom,*shortwordcom;
    char key,ans,chioce,tword,datacom,doscommmand[15];
    int input,div,i,a,b,d,inp,numcom,command;
    int y, maxcolor;
    float max,x;
    complex input_data[SAMPLES], rv_data[SAMPLES], fft_data[SAMPLES];
```

float amp[MAX\_HARMONIC];

สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
int gd, gm;
```

```
int bit;
```

```
clrscr();
```

```
do { for (i=0;i<Max_array;i++)
```

```
    { R[i]=0; }
```

```
input=0;
```

```
outportb(pctrl,0x91);
```

```
do { clrscr();
```

```
    /* choose command */
```

```
    printf("What do you want to work press choose your chioce\n");
```

```
    printf("    1. Write base sound\n");
```

```
    printf("    2. To command \n");
```

```
    printf("    3. To exit \n");
```

```
    printf("Press your chioce : ");
```

```
    chioce = getch();
```

```
    printf("\n");
```

```
    } while((chioce != '1') && (chioce != '2') && (chioce != '3'));
```

```
switch(chioce){
```

```
    case '1' : do { clrscr();
```

```
        printf("Dose word you want get is Old word or New word (Old:New) :
```

```
");
```

```
        tword =getch(); } while((tword != 'o') && (tword != 'n'));
```

```
        switch(tword) { case 'n' : clrscr();
```

```
            printf(" What's word you want get : ");
```

```
            gets(word);
```

```
            break;
```

```
        case 'o' : clrscr();
```

```
            printf(" What's word you want get : ");
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        gets(word);
        break;
    default : break;}

do { printf("press enter when ready\n");
    key=getch();
    } while(key!='\n');

/* GET DATA */
outportb(portc,0x10);
delay(500);
outportb(portc,0x00);
while (input != 0x0f)
    input=inportb(portc); /* RAM full */
outportb(portc,0x10); /* clear add of RAM */
for(i=0;i<20000;i++) /* receive data from RAM */
    { data[i]=inportb(porta);
      outportb(portc,0x20);
      outportb(portc,0x60);
    }

/* chang data */
max=50;q=0;a=1;in_i=1;
for(i=a;i<20000;i++)
    { if (max<=data[i])
      { max=data[i];
        in_i=i;
        i=20000;} }
for(i=in_i;i<20000;i++)
    { if (data[i]>max)
      c=i;
    }
q=c-in_i;
printf("Number'F = %d\n",q);
h=q/Max_array;
dif();

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

if (tword =='\n')
    { if (maxdif<=400)
        { if (( shortwordcom = fopen("shortcom.and","a")) == NULL) {
            printf("Error in open file'shortword
command\n");
            exit(1); }
        fprintf(shortwordcom,"%s\n",word);
        fclose(shortwordcom);
        }
    else { if (( longwordcom = fopen("longcomm.and","a")) == NULL) {
        printf("Error in open file longwordcommand");
        exit(1); }
        fprintf(longwordcom,"%s\n",word);
        fclose(longwordcom);
        }
    /* get random data */
    div=0;
    if (h==0)
        h=1;
    for (i=in_i;i<=c;i=i+h)
        { if (div!=Max_array)
            { if ( i!=c)
                { R[div]=data[i]/(51.2);
                div=div+1; }
            else break ; }
        else break; }

    sampling(input_data, SAMPLES);
    bit = log(SAMPLES)/log(2);
    bit_reverse(input_data, bit, rv_data);
    fft(rv_data, SAMPLES, fft_data);
    cal_amp(fft_data, SAMPLES, amp);
    grap(amp);

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

do { clrscr();
    printf("These're data do you want to compile (Yes:No) : ");
    datacom = getch(); } while((datacom != 'y') && (datacom != 'n'));
printf("\n");
switch(datacom) { case 'n' : exit(0);
                    break;
                    default : printf("Please wait.. \n");
                    break; }

if (maxdif>450)
    { savefile(); }
openfile(amp);
break;

case 'Z' : do { printf("press enter when ready\n");
                key=getch();
            } while(key!='\r');

/* GET DATA */
outportb(portc,0x10);
delay(500);
outportb(portc,0x00);
while (input != 0x0f)
    input=inportb(portc); /* RAM full */
outportb(portc,0x10); /* clear add of RAM */
for(i=0;i<20000;i++) /* receive data from RAM */
    { data[i]=inportb(porta);
      outportb(portc,0x20);
      outportb(portc,0x60);
    }

/* chang data */
max=50;q=0;a=1;in_i=1;
for(i=a;i<20000;i++)
    { if (max<=data[i])
      { max=data[i];

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        in_i=i;
        i=20000;} }
for(i=in_i;i<20000;i++)
    { if (data[i]>max)
        c=i;
    }
q=c-in_i;
printf("Number F = %d\n",q);
h=q/Max_array;

/* get random data */
div=0;
if (h==0)
    h=1;
for (i=in_i;j<=c;i=i+h)
    { if (div!=Max_array)
        { R[div]=data[i]/(51.2);
          div=div+1; }
      else break; }
dif();

sampling(input_data, SAMPLES);
bit = log(SAMPLES)/log(2);
bit_reverse(input_data, bit, rv_data);
fft(rv_data, SAMPLES, fft_data);
cal_amp(fft_data, SAMPLES, amp);

grap(amp);

command=filecomp(amp,SAMPLES);
printf(" %s ",wordcomp[command]);

if ( strcmp(wordcomp[command],"dir") == 0 )
    { strcat(wordcomp[command],"/w/p"); }
if ( strcmp(wordcomp[command],"chk") == 0 )

```

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ หากท่านต้องการนำเอกสารนี้ไปใช้โดยไม่ได้รับอนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        { strcat(wordcomp[command],"disk"); }
    if ( strcmp(wordcomp[command],"scan") == 0 )
        { strcat(wordcomp[command],"disk"); }
    if ( strcmp(wordcomp[command],"mem") == 0 )
        { strcat(wordcomp[command],"/c/p"); }

    /*      if (strcmp(wordcomp[command],"") != 0 )
            { system(wordcomp[command]); } */

        break;
    case '3' : exit(0);
        break;
    default : break; }

printf("Do you want to exit press 'n'");
ans=getch();
} while(ans!='n');
}

/* sampling : find value of each sample of square wave function. */
void sampling(complex *square_wave, int samples)
{
    int i;

    for (i=0;i<samples;i++)
    { square_wave[i].re=R[i];
      square_wave[i].im=0; }
    /*for (i = 0; i < samples/2; i++) {
        square_wave[i].re = AMPLITUDE_MAX;
        square_wave[i].im = 0;
    }
    for (i = samples/2; i < samples; i++) {
        square_wave[i].re = AMPLITUDE_MIN;
        square_wave[i].im = 0;
    }
}

```

เอกสารนี้เป็นทรัพย์สินทางปัญญาของมหาวิทยาลัยเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

```

/* bit_reverse : return bit-reversed value of num. */
int bit_reverse(complex *input_data, int bit, complex *rv_data)
{
    int i, j, right_bit, rv_addr, num, addr;

    num = pow(2, bit);
    for (i = 0; i < num; i++) {
        rv_addr = 0;
        addr = i;
        for (j = 0; j < bit; j++) {
            right_bit = addr%2;
            rv_addr = rv_addr + right_bit*pow(2, bit-1-j);
            addr = addr/2;
        }
        rv_data[rv_addr] = input_data[i];
    }
}

/* fft : calculate fft. */
void fft(complex *input_data, int samples, complex *fft_data)
{
    int i;

    dit_fft(input_data, samples, fft_data);
    for (i = 0; i < samples; i++) {
        fft_data[i].re = fft_data[i].re*2/samples;
        fft_data[i].im = fft_data[i].im*2/samples;
    }
}

/* dit_fft : proceed fft algorithm. */
void dit_fft(complex *input_data, int samples, complex *fft_data)
{
    int half_samples, i;

```

เอกสารนี้เป็นเอกสารที่สวอนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

complex temp1, temp2;

if (samples > 2) {
    half_samples = samples/2;
    dit_fft(input_data, half_samples, fft_data);
    dit_fft(input_data + half_samples, half_samples, fft_data + half_samples);
    for (i = 0; i < half_samples; i++) {
        temp1 = fft_data[i];
        W(samples, i, &temp2);
        complex_multiply(&temp2, temp2, fft_data[i + half_samples]);
        complex_add(&fft_data[i], temp1, temp2);
        complex_subtract(&fft_data[i + half_samples], temp1, temp2);
    }
}
else {
    complex_add(&fft_data[0], input_data[0], input_data[1]);
    complex_subtract(&fft_data[1], input_data[0], input_data[1]);
}
}

/* cal_amp : calculate amplitude of complex number, fft_data, and return in amp. */
void cal_amp(complex *fft_data, int samples, float *amp)
{
    int i;
    float pp, PP[5];
    pp=0;
    /* amp = square root of (re^2 + im^2). */
    for (i = 0; i < samples; i++)
        { amp[i] = sqrt(fft_data[i].re * fft_data[i].re + fft_data[i].im * fft_data[i].im);
        }
    for (i=5;i<samples/2;i++)
        { if (amp[i]>pp)
            { pp=amp[i];
              i[1]=i; }
        }
    for (i=0;i<=5;i++)

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

PP[i]=0;
for (i = 5;i<samples/2;i++)
{ if (amp[i]>PP[0] && amp[i]<pp)
  { PP[0]=amp[i];
    ii[2]=i; }
}
for (i = 5;i<samples/2;i++)
{ if (amp[i]>PP[1] && amp[i]<PP[0] && amp[i]<pp)
  { PP[1]=amp[i];
    ii[3]=i; }
}
for (i = samples/2;i<samples-5;i++)
{ if (amp[i]>PP[2] )
  { PP[2]=amp[i];
    ii[4]=i; }
}
for (i = samples/2;i<samples-5;i++)
{ if (amp[i]>PP[3] && amp[i]<PP[2] )
  { PP[3]=amp[i];
    ii[5]=i; }
}
for (i = samples/2;i<samples-5;i++)
{ if (amp[i]>PP[4] && amp[i]<PP[3] && amp[i]<PP[2] )
  { PP[4]=amp[i];
    ii[6]=i; }
}
}

```

```
/* W : return (e-j2PI/n1)n2 */
```

```
void W(int n1, int n2, complex *result)
```

```
{
  result->re = cos(2*PI*n2/n1);
```

```
  result->im = sin(-2*PI*n2/n1);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

/* complex_add : add 2 complex number. */
void complex_add(complex *result, complex adder1, complex adder2)
{
    result->re = adder1.re + adder2.re;
    result->im = adder1.im + adder2.im;
}

/* complex_subtract : subtract complex number, sub2, from sub1. */
void complex_subtract(complex *result, complex subtracter1, complex subtracter2)
{
    result->re = subtracter1.re - subtracter2.re;
    result->im = subtracter1.im - subtracter2.im;
}

/* complex_multiply : multiply 2 complex number. */
void complex_multiply(complex *result, complex multiplicand, complex multiplier)
{
    result->re = multiplicand.re*multiplier.re - multiplicand.im*multiplier.im;
    result->im = multiplicand.re*multiplier.im + multiplicand.im*multiplier.re;
}

void grap(float *amp)
{
    int i,inp;
    float re,x,y;

    Initialize_Graphics_Mode();
    x=100;y=300;
    setcolor(3);
    rectangle(35,0,600,300);
    setviewport(35,0,600,300,1);
    outtextxy(40,5, "Raw data");
    outtextxy(0,300,"0.0");
    outtextxy(0,44.0,"5.0");
    outtextxy(0,95.2,"4.0");
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

outtextxy(0,146.4,"3.0");
outtextxy(0,197.6,"2.0");
outtextxy(0,248.8,"1.0");
moveto(0,300);
setcolor(4);
for (i=in_i ; i <=c ; i++)
{
    imp = data[i] ;
    line(x,y,x+0.02,300-imp);
    x=x+0.02;
    y = 300-imp;
}
getchar();
clearviewport();
closegraph();
Initialize_Graphics_Mode();
x=0;y=239;
setcolor(3);
rectangle(0,0,639,479);
setviewport(0,0,639,479,1);
line(0,239,320,239);
for (i=1;i<=Max_array;i++)
    { re = amp[i]*150;
      setcolor(2);
      line(x,239,x,239-re);
      x=x+0.3;
    }
getchar();
clearviewport();
closegraph();
}

```

**void Initialize\_Graphics\_Mode(void)**

```

{ int gdriver = DETECT,gmode,errorcode;
  initgraph(&gdriver,&gmode,"");

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

errorcode=graphresult();
if (errorcode != grOk)
{ printf("Graphics error : %s\n",grapherrormsg(errorcode));
  printf("Press any key to halt : ");
  getch();
  exit(1);
}
}

```

```
void openfile(float *amp)
```

```

{ FILE *sumfr,*sdf,*summed,*sdmed,*sumend,*sdend,*frfile[6],*medfile[12],*endfile[6];
  FILE *infileim[7][10],*sumfileim,*sdfileim,*fwordp,*avgfwordp;
  int i,j,a,f,dummyf;
  float number,sumnum,sumnumi,avgfr[6],avgmed[12],avgend[6],SDfr[6],SDfri[6],SDmed[12];
  float SDmedi[12],SDend[6],SDendi[6],SDdummy,SDdummyi,sumf,avgf;
  float avg[7][10],avgi[7][10],SD[7][10],SDi[7][10],SDf;
  char sumfrword[15],sdfword[15],summedword[15],sdmedword[15],sumendword[15];
  char sdendword[15],frword[6][15],medword[12][15],endword[6][15],wordhelpim[7][9][15];
  char sumwordim[15],sdwordim[15],fword[15],avgfword[15];

  for (j=1;j<=6;j++){
    for(i=1;i<=9;i++){
      (
        strcpy(wordhelpim[j][i],word); )
    }
  }
  for (i=1;i<=5;i++)
  { strcpy(frword[i],word);
    strcpy(endword[i],word);
  }
  for (i=1;i<=11;i++)
  { strcpy(medword[i],word); }

  strcat(frword[1],"01.dat");
  strcat(frword[2],"02.dat");
  strcat(frword[3],"03.dat");

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

strcat(frword[4], "04.dat");

strcat(frword[5], "05.dat");

strcat(endword[1], "91.dat");

strcat(endword[2], "92.dat");

strcat(endword[3], "93.dat");

strcat(endword[4], "94.dat");

strcat(endword[5], "95.dat");

strcat(medword[1], "m1.dat");

strcat(medword[2], "m2.dat");

strcat(medword[3], "m3.dat");

strcat(medword[4], "m4.dat");

strcat(medword[5], "m5.dat");

strcat(medword[6], "m6.dat");

strcat(medword[7], "m7.dat");

strcat(medword[8], "m8.dat");

strcat(medword[9], "m9.dat");

strcat(medword[10], "m10.dat");

strcat(medword[11], "m11.dat");

strcat(wordhelpim[1][1], "11.dat");

strcat(wordhelpim[1][2], "12.dat");

strcat(wordhelpim[1][3], "13.dat");

strcat(wordhelpim[1][4], "14.dat");

strcat(wordhelpim[1][5], "15.dat");

strcat(wordhelpim[1][6], "16.dat");

strcat(wordhelpim[1][7], "17.dat");

strcat(wordhelpim[1][8], "18.dat");

strcat(wordhelpim[1][9], "19.dat");

strcat(wordhelpim[2][1], "21.dat");

strcat(wordhelpim[2][2], "22.dat");

strcat(wordhelpim[2][3], "23.dat");

strcat(wordhelpim[2][4], "24.dat");

strcat(wordhelpim[2][5], "25.dat");

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

strcat(wordhelpim[2][6],"26.dat");  
 strcat(wordhelpim[2][7],"27.dat");  
 strcat(wordhelpim[2][8],"28.dat");  
 strcat(wordhelpim[2][9],"29.dat");

strcat(wordhelpim[3][1],"31.dat");  
 strcat(wordhelpim[3][2],"32.dat");  
 strcat(wordhelpim[3][3],"33.dat");  
 strcat(wordhelpim[3][4],"34.dat");  
 strcat(wordhelpim[3][5],"35.dat");  
 strcat(wordhelpim[3][6],"36.dat");  
 strcat(wordhelpim[3][7],"37.dat");  
 strcat(wordhelpim[3][8],"38.dat");  
 strcat(wordhelpim[3][9],"39.dat");

strcat(wordhelpim[4][1],"41.dat");  
 strcat(wordhelpim[4][2],"42.dat");  
 strcat(wordhelpim[4][3],"43.dat");  
 strcat(wordhelpim[4][4],"44.dat");  
 strcat(wordhelpim[4][5],"45.dat");  
 strcat(wordhelpim[4][6],"46.dat");  
 strcat(wordhelpim[4][7],"47.dat");  
 strcat(wordhelpim[4][8],"48.dat");  
 strcat(wordhelpim[4][9],"49.dat");

strcat(wordhelpim[5][1],"51.dat");  
 strcat(wordhelpim[5][2],"52.dat");  
 strcat(wordhelpim[5][3],"53.dat");  
 strcat(wordhelpim[5][4],"54.dat");  
 strcat(wordhelpim[5][5],"55.dat");  
 strcat(wordhelpim[5][6],"56.dat");  
 strcat(wordhelpim[5][7],"57.dat");  
 strcat(wordhelpim[5][8],"58.dat");  
 strcat(wordhelpim[5][9],"59.dat");

strcat(wordhelpim[6][1],"61.dat");

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
 ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

strcat(wordhelpim[6][2],"62.dat");
strcat(wordhelpim[6][3],"63.dat");
strcat(wordhelpim[6][4],"64.dat");
strcat(wordhelpim[6][5],"65.dat");
strcat(wordhelpim[6][6],"66.dat");
strcat(wordhelpim[6][7],"67.dat");
strcat(wordhelpim[6][8],"68.dat");
strcat(wordhelpim[6][9],"69.dat");

```

```
for (i=1;i<=5;i++)
```

```

{ if (( frfile[i] = fopen(frword[i],"a") == NULL ) {
    printf("Error in open file %d \n",frword[i]);
    printf("\007");
    exit(1); }
a=i-1;
fprintf(frfile[i],"%f\n",amp[a]);
fclose(frfile[i]);
}

```

```
for (i=1;i<=5;i++)
```

```

{ if (( endfile[i] = fopen(endword[i],"a") == NULL ) {
    printf("Error in open file %s \n",endword[i]);
    printf("\007");
    exit(1); }
a=2049-i;
fprintf(endfile[i],"%f\n",amp[a]);
fclose(endfile[i]);
}

```

```
for (i=1;i<=11;i++)
```

```

{ if (( medfile[i] = fopen(medword[i],"a") == NULL ) {
    printf("Error in open file %s \n",medword[i]);
    printf("\007");
    exit(1); }
a=1023+i;
fprintf(endfile[i],"%f\n",amp[a]);
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

fclose(endfile[i]);
}

for (j=1;j<=6;j++)
{ for (i=1;i<=9;i++)
  { if (( infileim[j][i] = fopen(wordhelpim[j][i],"a") == NULL ) {
    printf("Error in open file %d \n",wordhelpim[j][i]);
    printf("\007");
    exit(1); }
  a=(ii[j]-5)+i;
  fprintf(infileim[j][i],"%fv",amp[a]);
  fprintf(infileim[j][i],"%d\n",a);
  fclose(infileim[j][i]);
}
}

```

```

strcpy(sumwordim,word);
strcpy(sdwordim,word);
strcpy(sumfrword,word);
strcpy(sdfirword,word);
strcpy(sumendword,word);
strcpy(sdendword,word);
strcpy(summedword,word);
strcpy(sdmedword,word);

strcat(sumfrword,"s0.dat");
strcat(sdfirword,"s00.dat");
strcat(sumendword,"s1.dat");
strcat(sdendword,"s01.dat");
strcat(sumwordim,"s.dat");
strcat(sdwordim,"sd.dat");
strcat(summedword,"m.dat");
strcat(sdmedword,"msd.dat");

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

for (i=1;i<=5;i++)
{ if (( frfile[i] = fopen(frword[i],"r") == NULL ) {
    printf("Error in open file %d \n",frword[i]);
    printf("\007");
    exit(1); }
a=0;sumnum=0;sumnumi=0;
while(!feof(frfile[i])) {
    fscanf(frfile[i],"%f",&number);
    sumnum=sumnum+number;
    a=a+1; }
avgfr[i]=sumnum/(a);
fclose(frfile[i]);

if (( frfile[i] = fopen(frword[i],"r") == NULL ) {
    printf("Error in open file %d \n",frword[i]);
    printf("\007");
    exit(1); }
sumnum=0;sumnumi=0;
while(!feof(frfile[i])) {
    fscanf(frfile[i],"%f",&number);
    sumnum=sumnum+(pow((number-avgfr[i]),2));
    }
fclose(frfile[i]);
SDdummy=sumnum/(a);
SDfr[i]=sqrt(SDdummy);
}
}

```

```

for (i=1;i<=5;i++)
{ if (( endfile[i] = fopen(endword[i],"r") == NULL ) {
    printf("Error in open file %d \n",endword[i]);
    printf("\007");
    exit(1); }
a=0;sumnum=0;sumnumi=0;
while(!feof(endfile[i])) {
    fscanf(endfile[i],"%f",&number);

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        sumnum=sumnum+number;

        a=a+1; }
avgend[i]=sumnum/(a);
fclose(endfile[i]);

if (( endfile[i] = fopen(endword[i],"r") == NULL ) {
    printf("Error in open file %d \n",endword[i]);
    printf("\007");
    exit(1); }

sumnum=0;sumnumi=0;
while(!feof(endfile[i])) {
    fscanf(endfile[i],"%f",&number);
    sumnum=sumnum+(pow((number-avgend[i]),2));
}
fclose(endfile[i]);
SDdummy=sumnum/(a);
SDend[i]=sqrt(SDdummy);
}

if (( sumfr =fopen(sumfrword,"w") == NULL) {
    printf("Error in open file sumfrword \n");
    exit(1); }
for (i=1;i<=5;i++)
    { fprintf(sumfr,"%f\n",avgfr[i]); }
fclose(sumfr);

if (( sumend =fopen(sumendword,"w") == NULL) {
    printf("Error in open file sumendword \n");
    exit(1); }
for (i=1;i<=5;i++)
    { fprintf(sumend,"%f\n",avgend[i]); }
fclose(sumend);

if (( sdfr =fopen(sdfword,"w") == NULL) {
    printf("Error in open file sdfword \n");
    exit(1); }

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

for (i=1;i<=5;i++)
    { fprintf(sdf,"%f\n",SDfr[i]); }
fclose(sdf);

if (( ssend =fopen(ssendword,"w") == NULL) {
    printf("Error in open file ssendword \n");
    exit(1); }

for (i=1;i<=5;i++)
    { fprintf(ssend,"%f\n",SDend[i]); }
fclose(ssend);

for (i=1;i<=11;i++)
    { if (( medfile[i] = fopen(medword[i],"r") == NULL ) {
        printf("Error in open file %d \n",medword[i]);
        printf("\007");
        exit(1); }
    a=0;sumnum=0;sumnumi=0;
    while(!feof(medfile[i])) {
        fscanf(medfile[i],"%f",&number);
        sumnum=sumnum+number;
        a=a+1; }
    avgmed[i]=sumnum/(a);
    fclose(medfile[i]);

    if (( medfile[i] = fopen(medword[i],"r") == NULL ) {
        printf("Error in open file %d \n",medword[i]);
        printf("\007");
        exit(1); }
    sumnum=0;sumnumi=0;
    while(!feof(medfile[i])) {
        fscanf(medfile[i],"%f",&number);
        sumnum=sumnum+(pow((number-avgmed[i]),2));
    }
    fclose(medfile[i]);
    SDdummy=sumnum/(a);
    SDmed[i]=sqrt(SDdummy);

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

}

if (( summed = fopen(summedword,"w") == NULL) {
    printf("Error in open file summedword \n");
    exit(1); }
for (i=1;i<=5;i++)
    { fprintf(summed,"%f\n",avgmed[i]); }
fclose(summed);

if (( sdmed = fopen(sdmedword,"w") == NULL) {
    printf("Error in open file sdmedword \n");
    exit(1); }
for (i=1;i<=5;i++)
    { fprintf(sdmed,"%f\n",SDmed[i]); }
fclose(sdmed);

for (j=1;j<=6;j++)
{ for (i=1;i<=9;i++)
    { if (( infileim[j][i] = fopen(wordhelpim[j][i],"r") == NULL ) {
        printf("Error in open file %d \n",wordhelpim[j][i]);
        printf("\007");
        exit(1); }
        a=0;sumnum=0;sumnumi=0;
        while(!feof(infileim[j][i])) {
            fscanf(infileim[j][i],"%f",&number);
            fscanf(infileim[j][i],"%d",&f);
            sumnum=sumnum+number;
            sumnumi=sumnumi+f;
            a=a+1; }
        avg[j][i]=sumnum/(a);
        avgi[j][i]=sumnumi/(a);
        fclose(infileim[j][i]);

        if (( infileim[j][i] = fopen(wordhelpim[j][i],"r") == NULL ) {
            printf("Error in open file %d \n",wordhelpim[j][i]);
            printf("\007");

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        exit(1); }
sumnum=0;sumnumi=0;
while(!feof(infileim[j][i])) {
    fscanf(infileim[j][i],"%f",&number);
    fscanf(infileim[j][i],"%d",&f);
    sumnum=sumnum+(pow((number-avg[j][i]),2));
    sumnumi=sumnumi+(pow((f-avg[j][i]),2));
}
fclose(infileim[j][i]);
SDdummy=sumnum/(a);
SDdummyi=sumnumi/(a);
SD[j][i]=sqrt(SDdummy);
SDi[j][i]=sqrt(SDdummyi);
}
}

if ((sumfileim =fopen(sumwordim,"w")) == NULL) {
    printf("Error in open file sumwordim \n");
    exit(1); }
for (j=1;j<=6;j++)
{ for (i=1;i<=9;i++)
{ fprintf(sumfileim,"%f\n",avg[j][i]);
  fprintf(sumfileim,"%f\n",avgi[j][i]); }
}
fclose(sumfileim);

if ((sdfileim =fopen(sdwordim,"w")) == NULL) {
    printf("Error in open file sdwordim \n");
    exit(1); }
for (j=1;j<=6;j++)
{ for (i=1;i<=5;i++)
{ fprintf(sdfileim,"%f\n".SD[j][i]);
  fprintf(sdfileim,"%f\n",SDi[j][i]); }
}
fclose(sdftr);

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

strcpy(fword,word);
strcat(fword,".dat");
if (( fwordp = fopen(fword,"a") == NULL ) {
    printf("Error in open file %d \n",fword);
    printf("\007");
    exit(1); }
fprintf(fwordp,"%d\n",q);
fclose(fwordp);

```

```

if (( fwordp = fopen(fword,"r") == NULL ) {
    printf("Error in open file %d \n",fword);
    printf("\007");
    exit(1); }

```

```
avgf=0;f=0;sumf=0;
```

```
while(!feof(fwordp))
```

```
{ fscanf(fwordp,"%d",&dummyf);
```

```
sumf=sumf+dummyf;
```

```
f=f+1;
```

```
};
```

```
fclose(fwordp);
```

```
avgf=sumf/(f);
```

```

if (( fwordp = fopen(fword,"r") == NULL ) {
    printf("Error in open file %d \n",fword);
    printf("\007");
    exit(1); }

```

```
sumf=0;
```

```
while(!feof(fwordp))
```

```
{ fscanf(fwordp,"%d",&dummyf);
```

```
sumf=sumf+(pow((dummyf-avgf),2));
```

```
};
```

```
fclose(fwordp);
```

```
SDdummy=sumf/(f);
```

เอก SDf= $\sqrt{SDdummy}$ ; วนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
 ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

strcpy(avgfword,word);
strcat(avgfword,"fa.dat");
if (( avgfwordp = fopen(avgfword,"w") == NULL ) {
    printf("Error in open file %d \n",avgfword);
    printf("\007");
    exit(1); }
fprintf(avgfwordp,"%fn",avgf);
fprintf(avgfwordp,"%fn",SDf);
fclose(avgfwordp);
printf("Compile \n");
}

```

```
int filecomp(float *amp,int samples)
```

```

{ FILE *infileim[6],*sdim[6],*wordcom,*ffile[6],*pri[6],*end[6],*sumfr[6],*sdf[6],*summed[6];
FILE *sdmed[6],*sumend[6],*sdend[6];
int' parity,para,lll,i,j,l,x,o,murcom,fr[6][6],med[6][12],endd[6][6],dummyim[6][7][10];
int dummyim1[6][7][10],sumdummyim[6],sumfrend[6],summedd[6],maxvaluetotol[6],x,y;
int fnum[6],prin[6],endn[6],sumpe[6];
float datacompim[7][10],datacompimi[7][10],redata,redataim,redataimi,SD,SDim,SDimi;
float fredata,perl,z,pp,PP[6];
char wordhelpim[6][15],sdhelp[6][15],sumfrword[6][15],summedword[6][15],sdmedword[6][15];
char sdfirword[6][15],sumendword[6][15],sdendword[6][15],fword[6][15],priw[6][15],endw[6][15];

for (i=0;i<=5;i++)
    { for (l=0;l<=6;l++)
        { for (j=0;j<=9;j++)
            { dummyim[i][l][j]=0;
              dummyim1[i][l][j]=0;
            }
        }
        fnum[i]=0;
        prin[i]=0;
        endn[i]=0;
        sumpe[i]=0;

```

เอกสารนี้เป็น for (l=0;l<=5;l++) ไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

    { fir[j][l]=0;
      endd[i][l]=0; }
  for (l=0;l<=11;l++)
    { med[i][l]=0; }
}
ll=0;

if ((wordcom = fopen("shortcom.and","r")) == NULL) {
  printf("Error in open file 'shortwordcommand\n");
  exit(1); }
numcom=1;
while(!feof(wordcom))
{ fscanf(wordcom,"%s",wordcomp[numcom]);
  numcom=numcom+1;
}
fclose(wordcom);

if (maxdif>400)
{ ll=50;
  for(i=3;i<=(numcom-2);i++)
  { strcpy(priw[i],wordcomp[i]);
    strcpy(endw[i],wordcomp[i]);

    strcat(priw[i],"pa.dat");
    strcat(endw[i],"ea.dat");

    if (( pri[i] = fopen(priw[i],"r")) == NULL ) {
      printf("Error in open file %s \n",priw[i]);
      printf("\007");
      exit(1); }

    fscanf(pri[i],"%f",&redata);
    fscanf(pri[i],"%f",&SD);
    fclose(pri[i]);

    if (PRI>redata-SD && PRI<redata+SD)

```

เอกสารนี้เป็นเอกสารที่ ( print[i]=30; ) การใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

if (( end[i] = fopen(endw[i],"r") == NULL ) {
    printf("Error in open file %s \n",endw[i]);
    printf("\007");
    exit(1); }
fscanf(end[i], "%f",&redata);
fscanf(pri[i], "%f",&SD);
fclose(end[i]);

if (END>redata-SD && END<redata+SD)
    { endn[i]=30; }
if (prin[i]==30 && endn[i]==30)
    { sumpe[i]=60; }
}
}

for (i=5;i<samples/2;i++)
    { if (amp[i]>pp)
        { pp=amp[i];
          ii[1]=i; }
    }
printf("%f %d\n",pp,ii[1]);
for (i=0;i<=5;i++)
    PP[i]=0;
for (i = 5;i<samples/2;i++)
    { if (amp[i]>PP[0] && amp[i]<pp)
        { PP[0]=amp[i];
          ii[2]=i; }
    }
printf("%f %d\n ",PP[0],ii[2]);
for (i = 5;i<samples/2;i++)
    { if (amp[i]>PP[1] && amp[i]<PP[0] && amp[i]<pp)
        { PP[1]=amp[i];
          ii[3]=i; }
    }
printf("%f %d\n ",PP[1],ii[3]);

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับกรใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้拿去ใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งยังมีให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

strcpy(sdmedword[i],wordcomp[i]);
strcpy(fword[i],wordcomp[i]); }

for(i=1;i<=(numcom-2);i++)
{
  strcat(wordhelpim[i],"s.dat");
  strcat(sdhelp[i],"sd.dat");
  strcat(sumfrword[i],"s0.dat");
  strcat(sdfword[i],"s00.dat");
  strcat(sumendword[i],"s1.dat");
  strcat(sdendword[i],"s01.dat");
  strcat(summedword[i],"m.dat");
  strcat(sdmedword[i],"msd.dat");
  strcat(fword[i],"fa.dat"); }

for(i=1;i<=(numcom-2);i++)
{
  if (( sumfr[i] = fopen(sumfrword[i],"r") ) == NULL ) {
    printf("Error in open file %s \n",sumfrword[i]);
    printf("\007");
    exit(1); }

  if (( sdfr[i] = fopen(sdfword[i],"r") ) == NULL ) {
    printf("Error in open file %s \n",sdfword[i]);
    printf("\007");
    exit(1); }

  for (j=1;j<=5;j++)
  {
    fscanf(sumfr[i],"%f",&redataim);
    fscanf(sdfr[i],"%f",&SDim);

    if (amp[j-1]<=redataim+SDim && amp[j-1]>=redataim-SDim)
      { fr[i][j]=2; }

  }

  fclose(sumfr[i]);fclose(sdfr[i]);
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

( if (( sumend[i] = fopen(sumendword[i],"r") == NULL ) {
    printf("Error in open file %s \n",sumendword[i]);
    printf("\007");
    exit(1); }

if (( ssend[i] = fopen(ssendword[i],"r") == NULL ) {
    printf("Error in open file %s \n",ssendword[i]);
    printf("\007");
    exit(1); }

for (j=1;j<=5;j++)
    { fscanf(sumend[i],"%f",&redataim);
      fscanf(ssend[i],"%f",&SDim);
      if (amp[2048-j]<=redata+SD && amp[2048-j]>=redata-SD)
          { endd[i][j]=2; }
    }

fclose(sumend[i]);fclose(ssend[i]);
}

for(i=1;i<=(numcom-2);i++)
    { if (( summed[i] = fopen(summedword[i],"r") == NULL ) {
      printf("Error in open file %s \n",summedword[i]);
      printf("\007");
      exit(1); }

if (( sdmed[i] = fopen(sdmedword[i],"r") == NULL ) {
    printf("Error in open file %s \n",sdmedword[i]);
    printf("\007");
    exit(1); }

for (j=1;j<=11;j++)
    { fscanf(summed[i],"%f",&redataim);
      fscanf(sdmed[i],"%f",&SDim);
      if (amp[1023+j]<=redata+SD && amp[1023+j]>=redata-SD)
          { med[i][j]=5; }
    }

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

}

```
fclose(summed[j]);fclose(sdmed[i]);
```

}

```
for(i=1;i<=(numcom-2);i++)
```

```
{ if (( infileim[i] = fopen(wordhelpim[i],"r") == NULL ) {
```

```
    printf("Error in open file %s \n",wordhelpim[i]);
```

```
    printf("\007");
```

```
    exit(1); }
```

```
if (( sdim[i] =fopen(sdhelp[i],"r") == NULL) {
```

```
    printf("Error in open file %s \n",sdhelp[i]);
```

```
    printf("\007");
```

```
    exit(1); }
```

```
for (l=1;l<=6;l++)
```

```
{ for(j=1;j<=9;j++)
```

```
{
```

```
    fscanf(infileim[i],"%f",&redataim);
```

```
    fscanf(infileim[i],"%f",&redataimi);
```

```
    fscanf(sdim[i],"%f",&SDim);
```

```
    fscanf(sdim[i],"%f",&SDimi);
```

```
/* if( datacompim[l][j]<=redataim+(SDim*2) && datacompim[l][j]>=redataim-(SDim*2))
```

```
{
```

```
    dummyim[i][l][j]=3;
```

```
} */
```

```
if( datacompim[l][j]<=redataim+(SDim) && datacompim[l][j]>=redataim-(SDim) &&
```

```
datacompimi[l][j]>=redataimi-(SDimi) && datacompim[l][j]<=redataimi+(SDimi) )
```

```
{
```

```
    dummyim1[i][l][j]=2;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

    }
}
}
fclose(mfileim[i]);fclose(sdim[i]);

if (( ffile[i] = fopen(fword[i],"r") ) == NULL ) {
    printf("Error in open file %s \n",fword[i]);
    printf("\007");
    exit(1); }

fscanf(ffile[i],"%f",&fdata);
fscanf(ffile[i],"%f",&SD);
fclose(ffile[i]);
if(q>fdata-SD && q<fdata+SD)
    { fnum[i]=50; }
}
sumdummyim[0]=0;
sumfrend[0]=0;
for(i=1;i<=5;i++)
    { y=0;x=0;o=0;
    sumdummyim[i]=0;
    sumfrend[i]=0;
    for (l=1;l<=7;l++)
        { for(j=1;j<=9;j++)
            { y=y+dummyim[i][l][j];
              y=y+dummyiml[i][l][j];
            }
        }
    }
for (l=1;l<=5;l++)
    { x=x+fr[i][l];
      x=x+endd[i][l];
    }
for (l=1;l<=11;l++)
    { o=o+rned[i][l]; }

summedd[i]=o;
sumdummyim[i]=y;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        sumfrend[i]=x;
    }
    tolot[0]=0;
    for(i=1;i<=5;i++)
        { tolot[i]=0;
          tolot[i]=sumdummyim[i]+sumfrend[i]+summedd[i]+fnum[i]+prin[i]+cndn[i]+sumpe[i]+Ul; }
    maxvalue=0;
    if (maxdif<450)
        { z=130 ;}
    else { z=130 ; }
    printf (".");

    clrscr();
    for(i=1;i<=(numcom-2);i++)
        { per1=tolot[i]/z*100;
          printf("Word ( %s ) is correct %f %\n",wordcomp[i],per1);
        }
    for(i=1;i<=5;i++)
        { if (tolot[i]>tolot[maxvalue])
          maxvalue=i;
        }
    if (maxvalue==0)
        { printf("Word's not in memory \n"); }
    else { printf("Word command is %s \n",wordcomp[maxvalue]);}
    return(maxvalue);
}

```

dif()

```

{ int i,a,b,db,maxd,dd[4],w;

maxdif=0;a=0;b=0;maxd=0;
for (i=0;i<=3;i++)
    dd[i]=0;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

for (i=1;i<=20000;i++)
{ if (i==0)
    dd[0]++;
  if (i==1)
    dd[1]++;
  if (i==2)
    dd[2]++;
  if (i==3)
    dd[3]++; }
for (i=0;i<=3;i++)
  if (dd[i]>maxd)
    { maxd=dd[i];
      w=i; }
for (i=in_i;i<=c;i++)
  { if(data[i]==w)
    { b=i;
      a=0;
      while(data[b]==w)
        { a=a+1;
          b=b+1;
        }
      db=i;
      i=b;
    }
    if(a>maxdif)
      { maxdif=a;
        PRI=db-in_i;
        END=c-b;
      }
  }
}

```



**savefile()**

```

(FILE *prifile,*priafile,*endfile,*cndafile;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
 int i,f;  
 ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
float SD,avg,SDdummy,Sum,Dum;
char priword[15],priaword[15],endword[15],endaword[15];
```

```
strcpy(priword,word);
strcpy(priaword,word);
strcpy(endword,word);
strcpy(endaword,word);
strcat(priword,"p.dat");
strcat(priaword,"pa.dat");
strcat(endword,"c.dat");
strcat(endaword,"ca.dat");
```

```
if (( prifile = fopen(priword,"a") == NULL ) {
    printf("Error in open file %s \n",priword);
    printf("\007");
    exit(1); }
fprintf(prifile,"%d\n",PRI);
fclose(prifile);
if (( prifile = fopen(priword,"r") == NULL ) {
    printf("Error in open file %s \n",priword);
    printf("\007");
    exit(1); }
f=0;Sum=0;
while(!feof(prifile))
    { fscanf(prifile,"%f",&Dum);
      Sum=Sum+Dum;
      f=f+1;
    }
fclose(prifile);
avg=Sum/(f);

if (( prifile = fopen(priword,"r") == NULL ) {
    printf("Error in open file %s \n",priword);
    printf("\007");
    exit(1); }
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

while(!feof(prifile))
    { fscanf(prifile,"%f",&Dum);
      Sum=Sum+(pow((Dum-avg),2));
    }
fclose(prifile);
SDdummy=Sum/(f);
SD=sqrt(SDdummy);

if (( priafile = fopen(priaword,"w") == NULL ) {
    printf("Error in open file %d \n",priaword);
    printf("\007");
    exit(1); }

fprintf(priafile,"%f\n",avg);
fprintf(priafile,"%f\n",SD);
fclose(priafile);

if (( endfile = fopen(endword,"a") == NULL ) {
    printf("Error in open file %s \n",endword);
    printf("\007");
    exit(1); }

fprintf(endfile,"%d\n",END);
fclose(endfile);
if (( endfile = fopen(endword,"r") == NULL ) {
    printf("Error in open file %s \n",endword);
    printf("\007");
    exit(1); }

f=0;Sum=0;
while(!feof(endfile))
    { fscanf(endfile,"%f",&Dum);
      Sum=Sum+Dum;
      f=f+1;
    }
fclose(prifile);
avg=Sum/(f);

```

```

printf("Error in open file %s \n",endword);
printf("\007");
exit(1); }
Sum=0;
while(!feof(endfile))
{ fscanf(endfile,"%f",&Dum);
Sum=Sum+(pow((Dum-avg),2));
}
fclose(endfile);
SDdummy=Sum/(f);
SD=sqrt(SDdummy);

if (( endafile = fopen(endaword,"w") == NULL ) {
printf("Error in open file %d \n",endaword);
printf("\007");
exit(1); }
fprintf(endafile,"%fn",avg);
fprintf(endafile,"%fn",SD);
fclose(endafile);
}

```

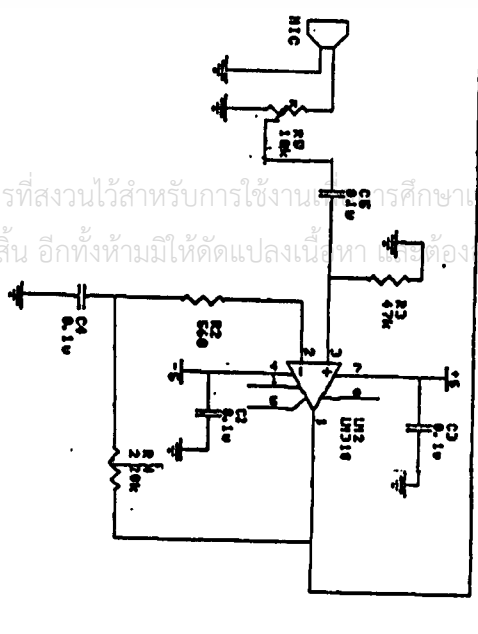
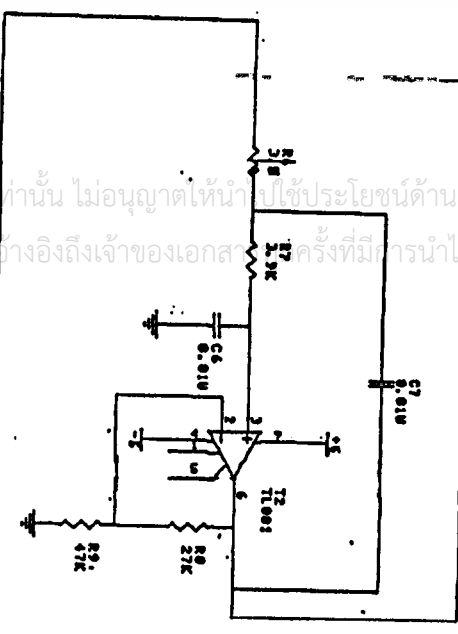
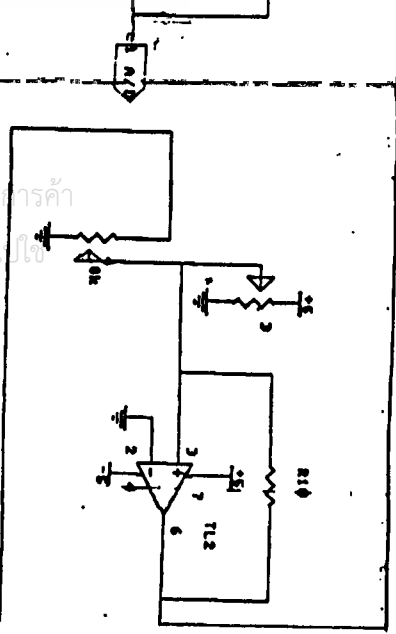
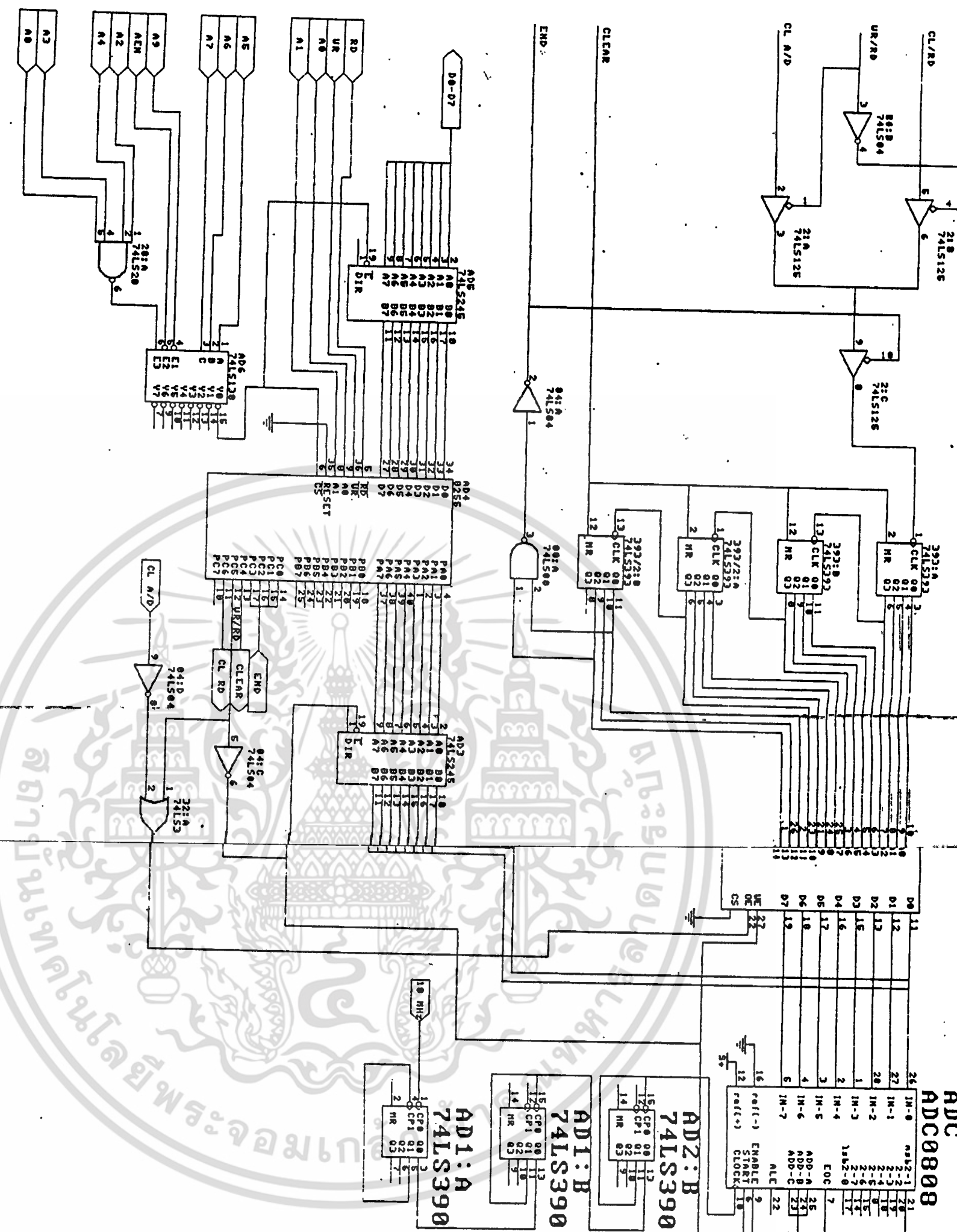


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

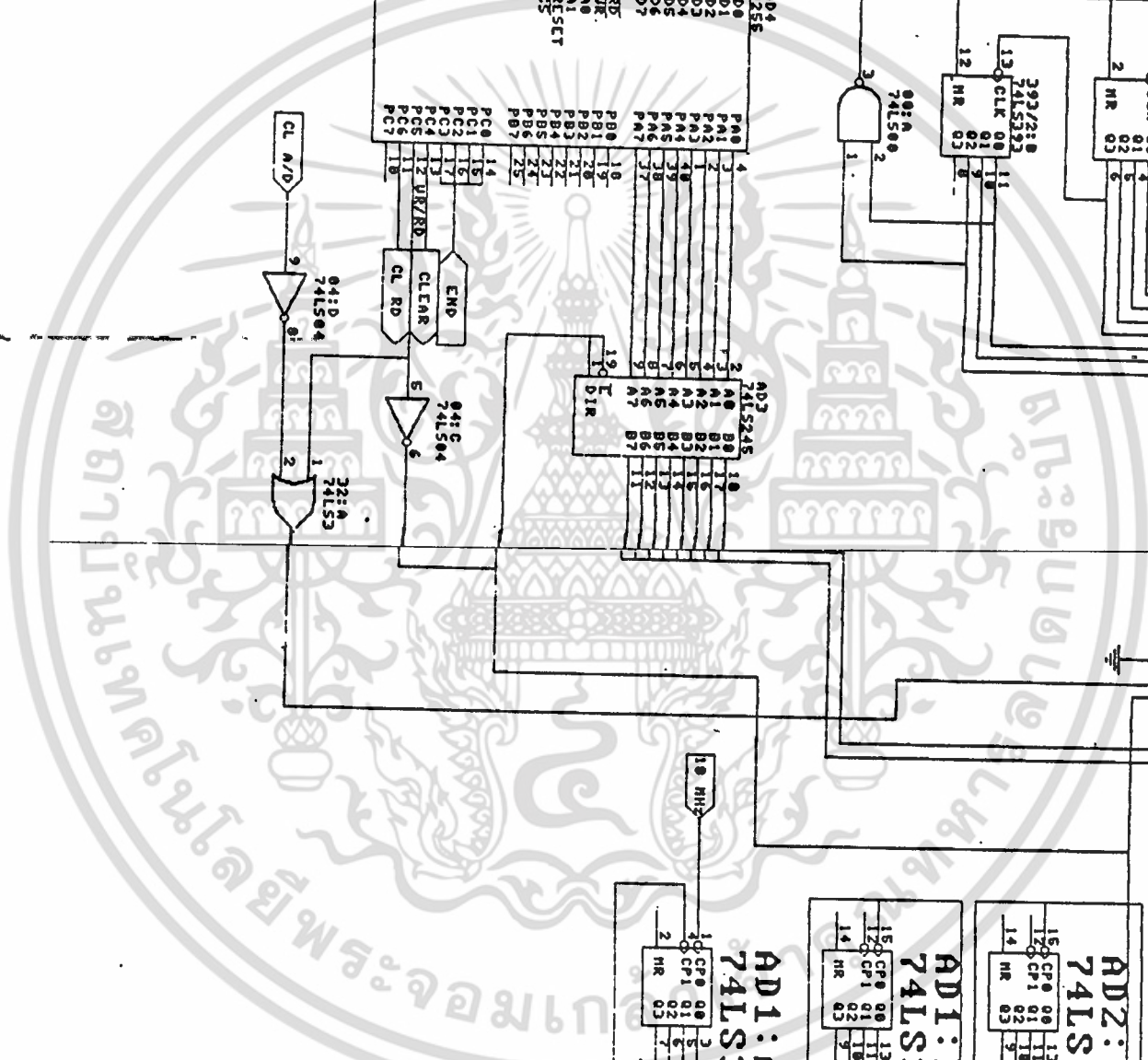


## ภาคผนวก 2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานในสถานศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
 ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



## หนังสืออ้างอิง

1. Shuzo Saito and Kazuo Nakata "Fundamentals of Speech Signal Processing" Academic Press;AP ,INC.New York 1985.
2. L.R. Rabiner and R.W. Schafer "Digital Processing of Speech Signals" Prentice-Hall.
3. Roman kue "Introduction to digital signal processing" McGraw Hill.
4. John P.Cater "Electronically Speaking : Computer Speech Generation"
5. Thomas W.Parsons "Voice and Speech Processing" McGraw Hill Book Company , New York 1987.
6. Lawrence Rabiner and Bing -Hwang Juang "Fundamentals of Speech Recognition" PTR : Prentice Hall Englewood Cliffs,New Jersey 1993.
7. Lawrence Rabiner and Ronald W.Schafer "Digital Signal Processing of Speech Signals" New York; Prentice Hall,1978.
8. Paul M.Emble and Bruce Kimble, "C Language Algorithms for Digital Signal Processing" New York; Prentice Hall,1991.
9. คร.ไพรัช ชัยพงษ์ "การประมวลผลสัญญาณดิจิทัลคอนการออกแบบวงจรดิจิทัล" NECTEC